sensors

MDPI

Article

An Anonymous User Authentication and Key Agreement Scheme Based on a Symmetric Cryptosystem in Wireless Sensor Networks

Jaewook Jung [1], Jiye Kim [1], Younsung Choi [2] and Dongho Won [1,*]

[1] Department of Computer Engineering, Sungkyunkwan University, 2066 Seoburo, Suwon, Gyeonggido 440-746, Korea; jwjung@security.re.kr (J.J.); jykim@security.re.kr (J.K.)

[2] Department of Cyber Security, Howon University, 64 Howon University 3 Gil, Impi-Myeon, Gunsan-Si, Jeonrabuk-Do 54058, Korea; yschoi@howon.ac.kr

* Correspondence: dhwon@security.re.kr; Tel.: +82-31-290-7213

Academic Editor: Leonhard M. Reind

Received: 2 June 2016; Accepted: 10 August 2016; Published: 16 August 2016

Abstract: In wireless sensor networks (WSNs), a registered user can login to the network and use a user authentication protocol to access data collected from the sensor nodes. Since WSNs are typically deployed in unattended environments and sensor nodes have limited resources, many researchers have made considerable efforts to design a secure and efficient user authentication process. Recently, Chen et al. proposed a secure user authentication scheme using symmetric key techniques for WSNs. They claim that their scheme assures high efficiency and security against different types of attacks. After careful analysis, however, we find that Chen et al.'s scheme is still vulnerable to smart card loss attack and is susceptible to denial of service attack, since it is invalid for verification to simply compare an entered ID and a stored ID in smart card. In addition, we also observe that their scheme cannot preserve user anonymity. Furthermore, their scheme cannot quickly detect an incorrect password during login phase, and this flaw wastes both communication and computational overheads. In this paper, we describe how these attacks work, and propose an enhanced anonymous user authentication and key agreement scheme based on a symmetric cryptosystem in WSNs to address all of the aforementioned vulnerabilities in Chen et al.'s scheme. Our analysis shows that the proposed scheme improves the level of security, and is also more efficient relative to other related schemes.

Keywords: wireless sensor networks; mutual authentication; key agreement; BAN-logic; smart card

1. Introduction

Wireless sensor networks (WSNs) are progressive ad hoc networks that are composed of quite a lot of resource-constrained sensor nodes that are randomly deployed over the target region [1]. Such networks provide cost-effective keys to a scope of monitoring problems, such as military battlefields, health care services, smart grid networks, and ubiquitous computing environments [2]. Moreover, the advanced technologies in the field of WSNs that a sensor attached to a device communicates with other ambient sensors are enabling to open the IoT environment. For these reasons, WSNs have been widely studied, both in the academic and industrial fields.

In WSNs, data gathered from sensor nodes sometimes include valuable and classified information such as details of the environmental surroundings during wartime, patient's private information, monitoring information of museums, and the voltage variation monitoring data in electric power companies. In order to ensure the confidentiality and reliability of deployed WSNs, it is important that access be allowed to registered and legitimate users only. In addition, secure protocol construction

positively requires a mutual authentication between an user and a sensor node. That is to say, a sensor node should be able to verify transmitted packet from a user to test a user's legitimacy. Meanwhile, a user also should be able to verify transmitted packet from a sensor node to test a normality of the sensor node. Besides, because of resource-constrained characteristics such as limited power, communication and computational capabilities [3], the mutual authentication and key agreement protocol should not be complex and resource consuming. For example, an asymmetric key cryptosystem, like RSA [4,5], ECC [6] or El-gamal [7,8], requires a high computational overhead that is unsuitable for the energy constraints of WSNs. Therefore, the authentication and key agreement protocols for WSNs should be designed to consider both security and efficiency perspectives.

1.1. Related Studies

In 1981, Lamport [9] first proposed a remote password authentication protocol for insecure channels, and since then, many authentication protocols have been studied [10–24], in order to enhance security and efficiency. In 2006, Wong et al. [10] proposed a password-based user authentication scheme with a light computational overhead using a one-way hash function and exclusive-OR operations. However, Tseng et al. [11] pointed out that Wong et al.'s scheme [10] could not resist replay and forgery attacks, and then proposed an enhanced scheme. They claimed that their scheme was secure against reply and forgery attacks, and that it provided improved efficiency in the password change process. In 2009, Vaidya et al. [12] described how neither the schemes provided by Wong et al. [10] and Tseng et al. [11] were secure against replay attacks and man-in-the-middle attacks. They also proposed a robust user authentication scheme for the WSN environment. In the same year, Das [13] proposed an enhanced authentication scheme as the basis for Wong et al.'s scheme [10]. He insisted that their scheme can resist different types of attacks, such as many logged-in-users with the same login identity attacks, off-line password guessing attacks, stolen-verifier attacks and impersonation attacks. However, Khan and Alghathbar [14] pointed out in 2010 that Das's scheme [13] could not resist gateway node bypassing attacks and privileged-insider attacks, and thus proposed an improved authentication scheme. In the same year, Vaidya et al. [15] demonstrated that Das's scheme [13] and Khan-Alghathbar's scheme [14] could not resist stolen smart card attacks, and they then proposed a enhanced version. Chen and Shih [16] also pointed out that Das scheme [13] could not resist parallel session attacks, and did not provide mutual authentication. In 2011, Fan et al. [17] proposed a user authentication protocol for two-tiered wireless sensor networks, and Yeh et al. [18] proposed an authentication protocol based on elliptic curves cryptography. In 2012, Das et al. [19] and Xue et al. [20] proposed a user authentication and key agreement scheme for WSNs based on the use of a smart card. These were both designed to fulfill various security requirements, such as key agreement, mutual authentication, password protection and prevention against several attacks. In 2014, Yuan [21] proposed an user authentication scheme based on biometric technique for WSNs. In the same year, Turkanović et al. [22] proposed a hash function based user authentication and key agreement protocol for heterogeneous ad hoc WSNs. They claimed that their scheme ensures a secure key agreement and mutual authentication and that it is also resilient against different types of attacks. However, Farash et al. [23] pointed out some security flaws in Turkanović et al.'s scheme [22], including a vulnerability to stolen-smart card attacks, man-in-the-middle attacks and sensor node impersonation attacks as well as the disclosure of secret parameters and the session key. They also suggested a user authentication and key agreement scheme for heterogeneous WSNs tailored for IoT environments. Recently, Amin et al. [24] demonstrated that Farash et al.'s scheme [23] could not resist stolen-smart card attacks, off-line password guessing attacks, user impersonation attacks, and known session-specific temporary information attacks, and proposed a improved version. Additionally, two-way authentication solutions on constraint devices using Datagram Transport Layer Security (DTLS) and Bellare-Canetti-Krawczyk (BCK) are proposed [25,26]. Porambage et al. [27] proposed an ECC-based authentication and key establishment scheme for WSNs in distributed IoT applications.

1.2. Motivations and Contributions

Chen et al. [28] recently suggested a secure user authentication scheme for wireless sensor networks. They claimed that their scheme could withstand different types of attacks, such as smart card loss attacks [29], replay attacks [30], stolen verifier attacks [31], privileged-insider attacks [32], user impersonation attacks [33], password guessing attacks [34], etc. They also claimed that their scheme was highly efficient, and very suited to WSN environments.

After performing a security analysis of Chen et al.'s scheme [28], however, we find that their scheme is still vulnerable to smart card loss attack, and is susceptible to denial of service attack, because it uses the incorrect verification method. In addition, we observe that their scheme cannot preserve user anonymity because user's identity is in plaintext form in login request message. Furthermore, their scheme cannot quickly detect an incorrect password during login phase, and this flaw wastes both communication and computational overheads.

In this paper, we describe how these attacks work, and propose an anonymous two-factor user authentication and key agreement scheme based on a symmetric cryptosystem in WSNs to address all of the previously mentioned problems regarding Chen et al.'s scheme [28].

1.3. The Threat Model

This subsection describes the threat model that we constructed with some common assumptions, including the capabilities of an attacker in WSNs environment.

(1) An attacker can control the communication channels between the user, gateway node, and sensor node, meaning that the attacker can intercept or modify any messages that are transmitted via the public channel [35,36].
(2) An attacker can modify and resend the intercepted/eavesdropped message [37].
(3) All of the existing smart cards are vulnerable, because the confidential information that is stored within them can be extracted by physically monitoring the power consumption [38], meaning that an attacker could read the data that is stored on a smart card.
(4) Due to the hostile environments in the deployment field, sensor nodes can be physically captured by an attacker. However, the gateway node is secure, meaning that an attacker cannot obtain the parameters from the gateway node [18,19].
(5) An attacker can easily guess low-entropy passwords and identities in an off-line manner, but the guessing of two secret parameters (e.g., password, identity) is computationally infeasible in polynomial time [39].

1.4. Security Requirements for User Authentication Scheme

A secure and efficient password-based user authentication scheme should fulfill some security requirements and defend some different types of attacks. In this subsection, we will examine the essential requirements of authentication scheme based on previous researches [9–24,28]. These requirements will be used to analyze the security of our proposed scheme in Section 5.

(1) User anonymity: A user's identity should be protected even if an attacker exploits user's smart card used for authentication scheme or if the messages which exchanged in communication group are exposed.
(2) Mutual authentication: Mutual authentication should be carried out between the user and gateway node, the gateway node and sensor node, and the sensor node and user, respectively.
(3) Session key agreement: The session key should be securely shared among other communication parties after the verification procedure is finished.
(4) Quick detection of the incorrect password: If a user enters the incorrect password by mistake in login phase, the password should be detected before performing verification phase.
(5) User friendliness: This property allows users to freely change/update their password without needing to communicate with the gateway node.

(6) Robustness: User authentication schemes should withstand different types of attacks.

- Smart card loss attacks: If an attacker steals a user's smart card, the attacker can extract the contents by the power consumption technique [38]. With obtained information, the attacker can try to launch various types of attacks.
- Off-line identity/password guessing attacks: An attacker tries to guess a identity/password and eventually find out the exact identity/password in an off-line environment by using the information stored in the smart card.
- User impersonation attacks: An attacker pretends to be the registered user with the forged login message by using the secret or public information that is collected from the smart cards and the data packets.
- Replay attacks: An attacker intercepts data packets for the purpose of making use of that data in some manner. Typically, this type of attack connotes copying and possibly modifying the data in various ways.
- Privileged-insider attacks: A privileged-insider attack literally means the attack mounted by a malicious insider. The malicious insiders have a noticeable advantage over external attackers because they have an authorized system admission and also may be familiar with the network design and system actions. Commonly, the malicious insiders want to obtain the users' private information such as their passwords.
- Denial of Service (DoS) attacks: A DoS attack is any event that diminishes or eliminates a network's capability of performing its expected function. In other words, an attacker mounts a DoS attack to make the server unavailable.
- Stolen-verifier attacks: An attacker steals a password-verifier from the gateway node and directly use it to masquerade as a legitimate user.
- Gateway node impersonation attacks: An attacker pretends to be the valid gateway node using the captured information.

1.5. Notations

All the notations mentioned in our proposed scheme and Chen et al.'s are specified in Table 1.

Table 1. Notations.

Value	Description
U_i	Remote user
S_n	Sensor node
GWN	Gateway node
ID_i, PW_i	Identity and password of U_i
SID_n	Identity of S_n
DID_i	Dynamic identity of U_i
k	The symmetric key
E_k, D_k	Encryption/Decryption with the symmetric key k
x_a	The secret parameter generated by the GWN, $(U_i \xleftrightarrow{x_a} GWN)$
x_s	The shared key between the GWN and S_n
$h(x_s\|\|SID_n)$	The secret key instead of x_s stored in S_n, $(GWN \xleftrightarrow{h(x_s\|\|SID_n)} S_n)$
b	A random number chosen by U_i
R_i	Cryptographic random numbers or nonces
$h(\cdot)$	One-way hash function
$X\|\|Y$	Concatenate operation
$\oplus$	XOR operation
T_1,T_2,T_3,T_4	Current timestamp
SK	Session key
ΔT	The maximum of transmission delay time

1.6. Organization of the Paper

The remainder of this paper is organized as follows: Section 2 reviews Chen et al.'s scheme, while Section 3 points out the weaknesses in Chen et al.'s scheme. Sections 4 and 5 present the

proposed scheme and the security analysis of the proposed scheme, respectively. Section 6 analyzes the performance of the proposed scheme in terms of the computational and communication costs; and lastly, Section 7 concludes the paper.

2. Review of Chen et al.'s Scheme

In this section, we describe Chen et al.'s authentication scheme [28]. Three communication parties comprise a user U_i, a gateway node GWN, and a sensor node S_n. This scheme is composed of four phases: registration, login, verification, and password change. We describe each phase in detail, and Figures 1–3 also illustrate Chen et al.'s scheme. Additionally, we describe the information on the sizes of all transmitted messages in the login and the verification phases. In order to compute the message size, based on [23], we set that both the block size of the symmetric encryption (E_k, D_k) and one-way hash function $h(\cdot)$ are 20 bytes long, the identity ID_i and password PW_i are 8 bytes, the random number b is 16 bytes, and the timestamp T_1–T_4 are 19 bytes long.

2.1. Registration Phase

(1) U_i selects ID_i and PW_i, and U_i then generates a random nonce b that is only known to the U_i. U_i computes a masked password $\overline{PW_i} = h(PW_i||b)$, and sends registration request message $\langle ID_i, \overline{PW_i} \rangle$ to GWN through a secure channel.

(2) GWN computes $N_i = h(ID_i||x_a) \oplus \overline{PW_i}$. GWN chooses a new smart card, and writes $\{ID_i, N_i, h(\cdot)\}$ into the smart card's memory. Then, GWN sends the smart card to U_i through a secure channel.

(3) U_i enters the random nonce b in its memory. Finally, the smart card contains the information $\{ID_i, N_i, h(\cdot), b\}$.

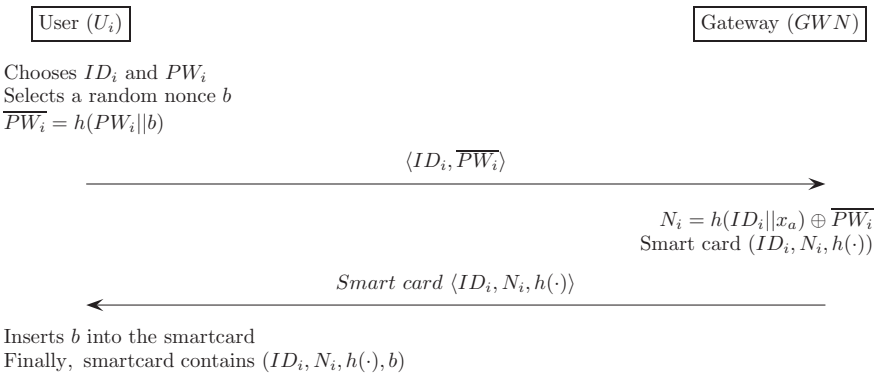

Figure 1. Registration phase for Chen et al.'s scheme.

2.2. Login Phase

(1) U_i inserts U_i's smart card into a terminal, and inputs the ID_i and PW_i. The smart card compares ID_i with the stored value ID_i. If this condition is satisfied, the smart card acknowledges the legitimacy of the U_i, and proceeds with the next step. Otherwise, it terminates this phase.

(2) The smart card computes $\overline{PW_i} = h(PW_i||b)$ and $k = h((N_i \oplus \overline{PW_i})||T_1)$, then chooses a random nonce $R_1 \in \{0,1\}^l$, and computes $A_i = E_k(ID_i||R_1||T_1)$.

(3) Finally, U_i sends a login request message $\langle ID_i, A_i, T_1 \rangle$ to GWN through a public channel.

From the above descriptions, in login phase of Chen et al.'s scheme, the message size of the login request $\langle ID_i, A_i, T_1 \rangle$ can be computed as (8 + 20 + 19) = 47 bytes.

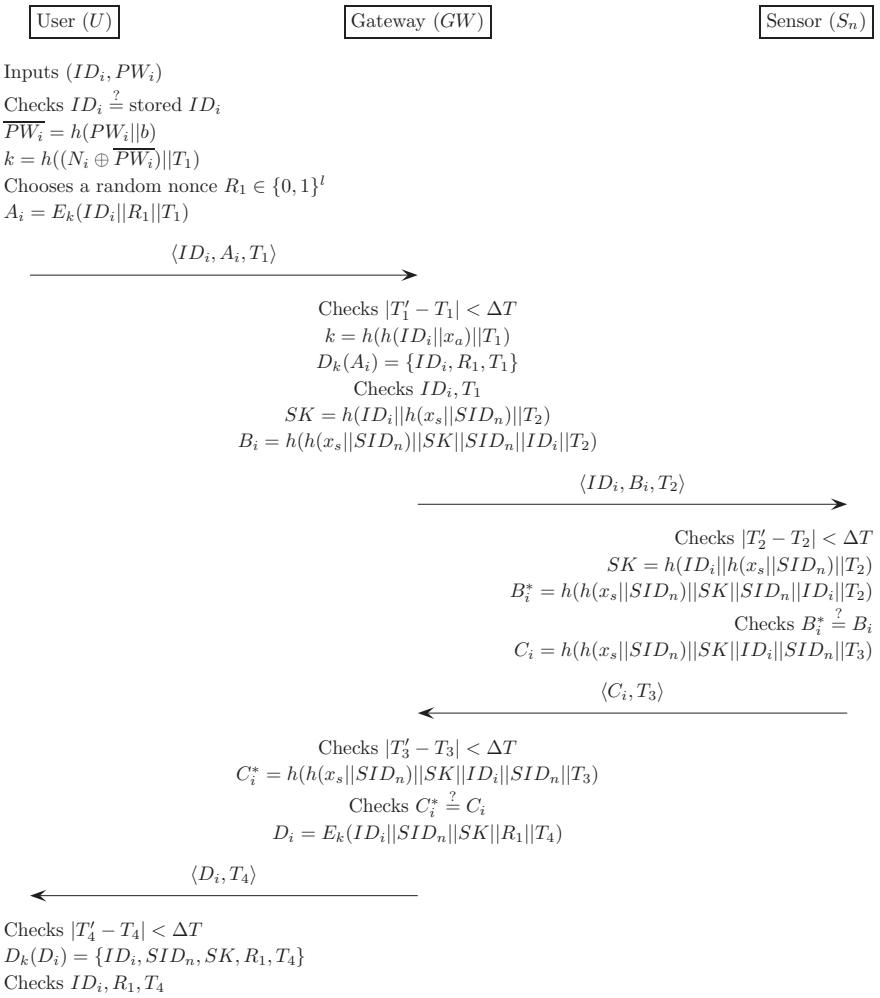

Figure 2. Login and verification phase for Chen et al.'s scheme.

2.3. Verification Phase

(1) *GWN* first checks the validity of the time-stamp $|T_1' - T_1| < \Delta T$. *GWN* computes $k = h(h(ID_i||x_a)||T_1)$ and decrypts $D_k(A_i) = \{ID_i, R_1, T_1\}$. *GWN* then compares ID_i and T_1 with the received values. If this condition is satisfied, *GWN* acknowledges the legitimacy of the U_i and proceeds with the next step. Otherwise, it terminates this phase.

(2) *GWN* computes $SK = h(ID_i||h(x_s||SID_n)||T_2)$ and $B_i = h(h(x_s||SID_n)||SK||SID_n||ID_i||T_2)$, then sends the message $\langle ID_i, B_i, T_2 \rangle$ to S_n through a public channel.

(3) S_n checks whether $|T_2' - T_2| < \Delta T$. S_n then computes $SK = h(ID_i||h(x_s||SID_n)||T_2)$ and $B_i^* = h(h(x_s||SID_n)||SK||SID_n||ID_i||T_2)$. S_n compares B_i^* with the received value B_i. If this condition is satisfied, S_n believes that the *GWN* is authentic. Otherwise, it terminates this phase.

(4) S_n computes $C_i = h(h(x_s||SID_n)||SK||ID_i||SID_n||T_3)$, and then sends the message $\langle C_i, T_3 \rangle$ to *GWN* through a public channel.

(5) *GWN* checks whether $|T_3' - T_3| < \Delta T$. *GWN* then computes $C_i^* = h(h(x_s||SID_n)||SK||ID_i||SID_n||T_3)$, and compares it with the received value C_i. If true, *GWN* believes that the S_n is authentic. Otherwise, *GWN* terminates this phase.

(6) *GWN* computes $D_i = E_k(ID_i||SID_n||SK||R_1||T_4)$, and sends the message $\langle D_i, T_4 \rangle$ to U_i through a public channel.

(7) U_i checks whether $|T_4' - T_4| \leq \Delta T$. U_i decrypts $D_k(D_i) = \{ID_i, SID_n, SK, R_1, T_4\}$ and compares ID_i, R_1 and T_4 with the previous values. If the verification does not hold, this phase is terminated. Otherwise, the U_i believes that the *GWN* is authentic, and successfully ends the verification phase.

From the above descriptions, in verification phase of Chen et al.'s scheme, the message size of the $\langle ID_i, B_i, T_2 \rangle$, $\langle C_i, T_3 \rangle$, and $\langle D_i, T_4 \rangle$ can be computed as (8 + 20 + 19) = 47 bytes, (20 + 19) = 39 bytes, and (20 + 19) = 39 bytes, respectively.

2.4. Password Change Phase

(1) U_i inserts U_i's smart card into a terminal and inputs ID_i, the old password PW_i and new password PW_i^*. The smart card compares the entered value ID_i with the ID_i stored in the smart card. If this condition is not satisfied, it terminates this phase. Otherwise, the smart card proceeds with the next step.
(2) The smart card computes $\overline{PW_i} = h(PW_i||b)$, $\overline{PW_i}^* = h(PW_i^*||b)$ and $N_i^* = N_i \oplus \overline{PW_i} \oplus \overline{PW_i}^*$.
(3) The smart card replaces the existing value N_i with the new value N_i^*. Finally, the smart card contains the information $\{ID_i, N_i^*, h(\cdot), b\}$.

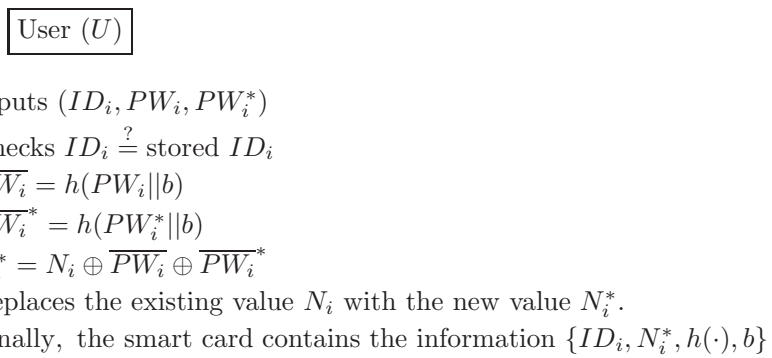

Figure 3. Password change phase for Chen et al.'s scheme.

3. Security Weaknesses of Chen et al.'s Scheme

In this section, we analyze the security of Chen et al.'s scheme [28]. Chen et al. claim that the scheme can withstand different types of attacks; however, based on attacker capabilities in Section 1.3, we found that their scheme is still vulnerable to smart card loss attack, and is also susceptible to denial of service attack, because it uses the incorrect verification method. In addition, we found that their scheme cannot preserve user anonymity. Since user's identity included in a login request message is in plain-text form when it transmitted to GW-node in login phase. In detail, user's identity on a public channel can be easily exposed to attackers, because they are able to eavesdrop on a public channel, as mentioned in Section 1.3. Furthermore, Chen et al.'s scheme missed a verification process to test input password, which led to the inefficiency problem. Since it is not able to detect an incorrect password during login phase, the login request message composed of incorrectly entered password sends to GW-node, and then GW-node detects the wrong message while performing a checking process on the login request message. Generally, the verification on the input password is recommended to perform immediately in login phase to avoid inefficiency problem [40]. We now describe the detailed weaknesses of Chen et al.'s schemes.

3.1. Smart Card Loss Attack

Suppose the smart card of U_i is stolen by the attacker, who extracts the stored secret values $\{ID_i, N_i, h(\cdot), b\}$ through physically monitoring the power consumption [38] as described in Section 1.3. With this information, the attacker can successfully lead to following malicious scenarios.

Scenario 1: If the attacker obtains the smart card, he or she can easily expose a user's identity ID_i through physically monitoring the power consumption [38]. Disclosure of the user's identity ID_i may allow tracking of the U_i's behavior and his or her current location.

Scenario 2: Using obtained smart card, the attacker can successfully pass the checking process of the login phase through using the ID_i in the smart card, because their checking process just compares the entered ID_i with the stored ID_i in the smart card. The same situation also happens for the password change phase.

Therefore, Chen et al.'s scheme still suffers from smart card loss attack.

3.2. Denial of Service Attack

When the attacker steals the user's smart card, the attacker can obtain the user's identity ID_i through physically monitoring the power consumption [38]. Through using this, in the password change phase, the attacker can easily set a new password, since it is invalid for verification to simply compare an entered ID_i and a stored ID_i in smart card. The following is a detailed description:

Step 1. The attacker inserts the U_i's smart card into a terminal, and enters the ID_i, PW_a and PW_a^*, where PW_a and PW_a^* are the attacker's arbitrary new passwords.
Step 2. The smart card compares the entered value ID_i with the ID_i stored in the smart card. At this time, it is obvious that this verification process turns out to be successful, since the entered ID_i is the same as the stored one in the smart card.
Step 3. The smart card computes $\overline{PW_a} = h(PW_a||b)$, $\overline{PW_a}^* = h(PW_a^*||b)$ and $N_a = N_i \oplus \overline{PW_a} \oplus \overline{PW_a}^*$.
Step 4. The smart card successfully replaces N_i with the new value N_a.

If an attacker stole the U_i's smart card and changed the password to an arbitrary new password as described above steps, then succeeding login requests by the legal user U_i will be rejected, unless they re-register with the GWN again. Therefore, Chen et al.'s scheme is vulnerable to a denial of service attack.

3.3. Failure to Preserve User Anonymity

User anonymity is a highly desirable requirement for user authentication schemes, because of the leakage of user's identity may allow an unauthorized entity to track the user's login record and behavior pattern. However, Chen et al.'s scheme states that a user's identity ID_i is in plaintext form during the login and verification phase. As described in Section 1.3, using an eavesdropping attack, the attacker can maliciously monitor the public channels [35,36], and also identify some of the valuable information in messages transmitted over these public channels.

In this manner, an attacker can without difficulty eavesdrop on login messages to collect the plaintext identities of communicating users. All of the eavesdropped messages can be analyzed by the attacker to track down the connections among the U_i, GWN and S_n, and for this reason, user anonymity cannot be preserved in Chen et al.'s proposal [28].

3.4. Incorrect Password Cannot be Quickly Detected

During the login phase of Chen et al.'s scheme [28], if the U_i inputs his/her identity and password, the smart card does not verify the validity of the U_i's password; therefore, if the U_i inputs an incorrect password by mistake, the login and verification phases are still carried out until they have been checked by GWN, leading to unnecessary communication and computational costs. The following detailed scenario explains this further.

Assume that the U_i inputs the ID_i and incorrect password PW_i^* during the login phase; the smart card then computes the following:

$$\begin{aligned} \overline{PW_i}^* &= h(PW_i^*||b) \\ k^* &= h((N_i \oplus \overline{PW_i}^*)||T_1) \\ R_1 &\in \{0,1\}^l \\ A_i^* &= E_{k^*}(ID_i||R_1||T_1) \end{aligned}$$

U_i sends a login request message $\langle ID_i, A_i^*, T_1 \rangle$ to GWN through a public channel. After receiving the login request message, GWN checks the validity of the time-stamp $|T_1' - T_1| < \Delta T$. GWN computes $k = h(h(ID_i||x_a)||T_1)$ and tries to decrypt $D_k(A_i^*) = \{ID_i, R_1, T_1\}$. GWN then compares ID_i and T_1 with the received values. If this comparison is satisfied, the GWN believes that the U_i is authentic. If not, it rejects the login request. However, it is obvious that GWN cannot decrypt $D_k(A_i^*)$, since k^* is not equal to k. Therefore, GWN belatedly realizes that entered password PW_i^* is an incorrect value, and GWN then terminates this procedure.

4. The Proposed Scheme

In this section, we propose an anonymous two-factor user authentication and key agreement scheme based on a symmetric cryptosystem in WSNs that addresses the security vulnerabilities in Chen et al.'s scheme [28]. Our proposed scheme also consists of the following four phases: registration, login, verification, and password change. We describe each phase in detail, and also describe the information on the sizes of all transmitted messages in the login and the verification phases. Table 1 summarizes the notation for the proposed scheme.

4.1. Registration Phase

The user registration phase begins when the U_i sends a registration request with his/her identity and a hashed password to GWN. The GWN then issues a smart card that stores some information, and sends it to U_i as a response to the registration request. The following describes this process in detail, and Figure 4 illustrates the registration phase for our proposed scheme.

(1) U_i selects ID_i and PW_i, and U_i then generates a random nonce b, that is only known to the U_i. U_i computes a masked password $\overline{PW_i} = h(PW_i||b)$ and sends registration request message $\langle ID_i, \overline{PW_i} \rangle$ to GWN through a secure channel.
(2) GWN computes $v = h(x_a)$, $N_i = h(ID_i||\overline{PW_i}) \oplus v$ and $M_i = h(\overline{PW_i}||v)$, and stores the v into the database. GWN then chooses a new smart card and writes $\{N_i, M_i, h(\cdot)\}$ into the smart card memory. After that the GWN sends the smart card to U_i through a secure channel.
(3) Upon receiving the smart card, U_i enters the random nonce b in its memory. Finally, the smart card contains the information $\{N_i, M_i, h(\cdot), b\}$.

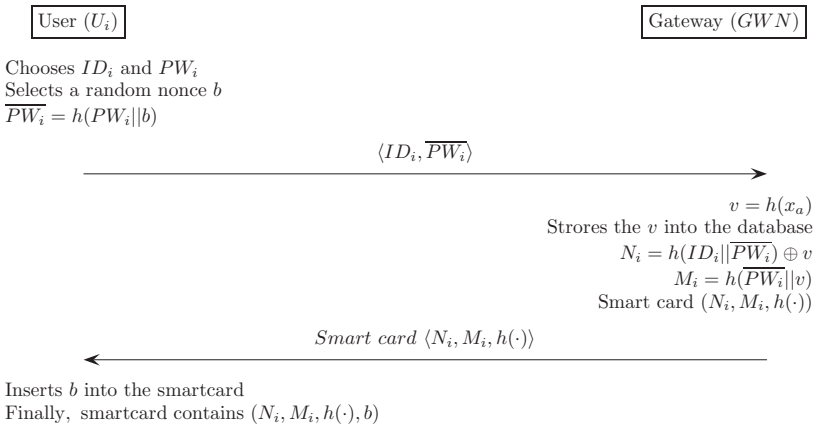

Figure 4. Registration phase for the proposed scheme.

4.2. Login Phase

The login phase is executed whenever the U_i wants to gain access to WSN. In this phase, U_i sends the login request to GWN. Figure 5 illustrates the login and verification phase for our proposed scheme. In detail, this process is:

(1) U_i inserts U_i's smart card into a terminal, and inputs the ID_i and PW_i. The smart card computes the masked password $\overline{PW_i}^* = h(PW_i||b)$ and $v^* = N_i \oplus h(ID_i||\overline{PW_i}^*)$. The smart card further computes $M_i^* = h(\overline{PW_i}^*||v^*)$, and compares it with the stored value M_i. If this condition is satisfied, the smart card acknowledges the legitimacy of the U_i, and proceeds with the next step. Otherwise, it terminates this phase.
(2) The smart card chooses a random nonce $R_1 \in \{0,1\}^l$, and computes $DID_i = h(ID_i||R_1)$. The smart card then computes $k = h(DID_i||v^*||T_1)$ and $A_i = E_k(DID_i||R_1||T_1)$.
(3) Finally, U_i sends a login request message $\langle DID_i, A_i, T_1\rangle$ to GWN through a public channel.

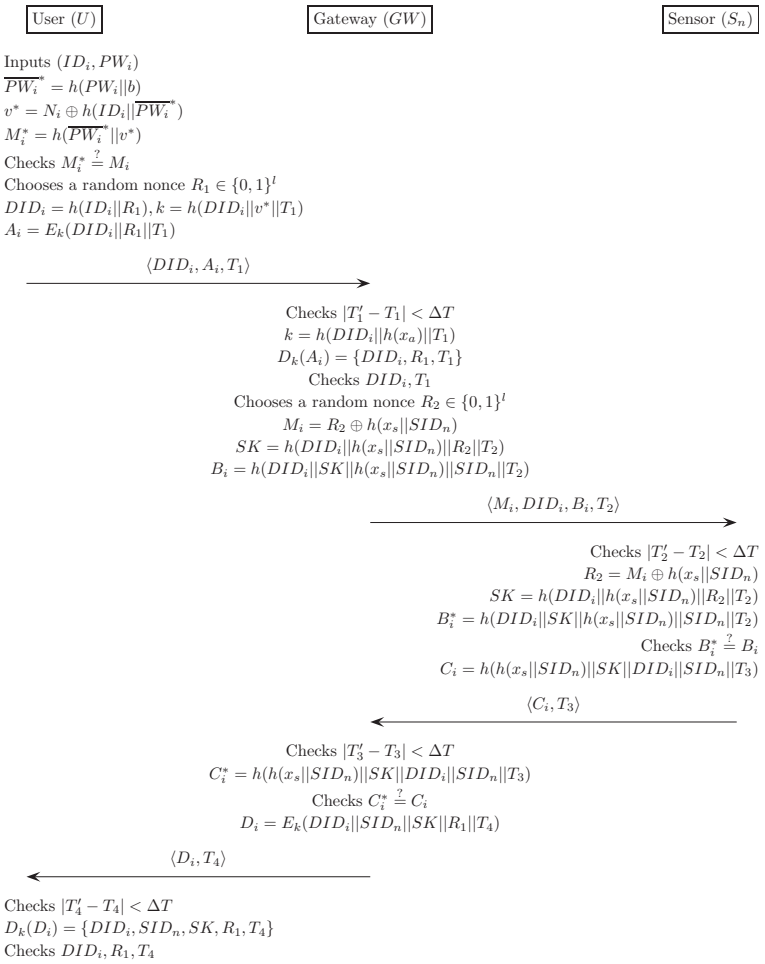

Figure 5. Login and verification phase for the proposed scheme.

From the above descriptions, in login phase of our propose scheme, the message size of the login request $\langle DID_i, A_i, T_1\rangle$ can be computed as (8 + 20 + 19) = 47 bytes.

4.3. Verification Phase

This phase executes several steps to achieve mutual authentication which is to test all transmitted message for judging the legitimacies of a U_i, GWN, and sensor node. As well as a session key agreement between all parties involved within the network. When GWN receives the login request message from the U_i, the verification phase begins. The following describes this process in detail.

(1) GWN first checks the validity of the time-stamp $|T_1' - T_1| < \Delta T$. GWN computes $k = h(DID_i||h(x_a)||T_1)$ and decrypts $D_k(A_i) = \{DID_i, R_1, T_1\}$. GWN then compares DID_i and T_1 with the received values. If this condition is satisfied, GWN acknowledges the legitimacy of the U_i and proceeds with the next step. Otherwise, it terminates this phase.

(2) GWN chooses $R_2 \in \{0,1\}^l$, and computes $M_i = R_2 \oplus h(x_s||SID_n)$. GWN further computes $SK = h(DID_i||h(x_s||SID_n)||R_2||T_2)$ and $B_i = h(DID_i||SK||h(x_s||SID_n)||SID_n||T_2)$, and then sends the message $\langle M_i, DID_i, B_i, T_2\rangle$ to S_n through a public channel.

(3) S_n first checks whether $|T_2' - T_2| < \Delta T$. If this condition does not hold, this phase is terminated. Otherwise, it computes $R_2 = M_i \oplus h(x_s||SID_n)$ and $SK = h(DID_i||h(x_s||SID_n)||R_2||T_2)$. The S_n further computes $B_i^* = h(DID_i||SK||h(x_s||SID_n)||SID_n||T_2)$ and compares it with the received value B_i. If this condition is satisfied, S_n believes that the GWN is authentic. Otherwise, it terminates this phase.

(4) S_n computes $C_i = h(h(x_s||SID_n)||SK||DID_i||SID_n||T_3)$, and then sends the message $\langle C_i, T_3\rangle$ to GWN through a public channel.

(5) GWN first checks whether $|T_3' - T_3| < \Delta T$. If the relationship does not hold, this phase is terminated. Otherwise, it computes $C_i^* = h(h(x_s||SID_n)||SK||DID_i||SID_n||T_3)$, and compares it with the received value C_i. If true, GWN believes that the S_n is authentic. Otherwise, it terminates this phase.

(6) GWN computes $D_i = E_k(DID_i||SID_n||SK||R_1||T_4)$, and sends the message $\langle D_i, T_4\rangle$ to U_i through a public channel.

(7) U_i first checks whether $|T_4' - T_4| \leq \Delta T$. If the relationship does not hold, it terminates this phase. Otherwise, it computes $D_k(D_i) = \{DID_i, SID_n, SK, R_1, T_4\}$, and compares DID_i, R_1 and T_4 with the previous values. If the verification does not hold, it terminates this phase. Otherwise, the U_i believes that GWN is authentic, and successfully ends the verification phase.

From the above descriptions, in verification phase of our proposed scheme, the message size of the $\langle M_i, DID_i, B_i, T_2\rangle$, $\langle C_i, T_3\rangle$, and $\langle D_i, T_4\rangle$ can be computed as (20 + 20 + 20 + 19) = 79 bytes, (20 + 19) = 39 bytes, and (20 + 19) = 39 bytes, respectively.

4.4. Password Change Phase

The password change phase is invoked whenever the U_i wants to change his or her old password to a new password. In the password change phase of our proposed scheme, U_i communicates without any assistance from the GWN. Figure 6 illustrates the password change phase for our proposed scheme. We now describe this process in further detail:

User (U)

Inputs $(ID_i, PW_i^{old}, PW_i^{new})$
$\overline{PW_i}^{old} = h(PW_i^{old}||b), v^{old} = N_i \oplus h(ID_i||\overline{PW_i}^{old})$
$M_i^{old} = h(\overline{PW_i}^{old}||v^{old})$
Checks $M_i^{old} \stackrel{?}{=} M_i$
$\overline{PW_i}^{new} = h(PW_i^{new}||b)$
$N_i^{new} = h(ID_i||\overline{PW_i}^{new}) \oplus v, M_i^{new} = h(\overline{PW_i}^{new}||v)$
Replaces the existing values N_i and M_i
with the new values N_i^{new} and M_i^{new}, respectively.
Finally, the smart card contains the information $\{N_i^{new}, M_i^{new}, h(\cdot), b\}$

Figure 6. Password change phase for the proposed scheme.

(1) U_i inserts U_i's smart card into a terminal, and inputs ID_i, old password PW_i^{old}, and new password PW_i^{new}. The smart card computes the old masked password $\overline{PW_i}^{old} = h(PW_i^{old}||b)$, $v^{old} = N_i \oplus h(ID_i||\overline{PW_i}^{old})$, and $M_i^{old} = h(\overline{PW_i}^{old}||v^{old})$. The smart card then verifies whether $M_i = M_i^{old}$. If this condition is not satisfied, it terminates this phase. Otherwise, the smart card proceeds with the next step.

(2) The smart card computes $\overline{PW_i}^{new} = h(PW_i^{new}||b)$, $N_i^{new} = h(ID_i||\overline{PW_i}^{new}) \oplus v$ and $M_i^{new} = h(\overline{PW_i}^{new}||v)$
(3) The smart card replaces the existing values N_i and M_i with the new values N_i^{new} and M_i^{new}, respectively. Finally, the smart card contains the information $\{N_i^{new}, M_i^{new}, h(\cdot), b\}$.

5. Security Analysis and Proof of the Proposed Scheme

In this section, we present a security analysis of our proposed scheme. We first examine whether our proposed scheme is safe, and we also consider its ability to resist various known attacks as described in Section 1.4. Then we adopt Burrows-Abadi-Needham (BAN) logic [41] to prove that a session key can be correctly generated between U_i, GWN and S_n.

5.1. Security Analysis of the Proposed Scheme

In this subsection, we scrutinize whether our proposed scheme can not only withstand various attacks, but also satisfy basic requirements that the security scheme claims. Moreover, we conduct a comparative analysis [13–20,28], which describes in Table 2. Details of the results are illustrated below.

Table 2. Security comparison of our proposed scheme and other related schemes.

Features	Das et al. [13]	K-A- [14]	Vaidya et al. [15]	C-S- [16]	Fan et al. [17]	Yeh et al. [18]	Das et al. [19]	Xue et al. [20]	Chen et al. [28]	Proposed Scheme
Proposition 1	×	√	×	√	√	×	×	×	×	√
Proposition 2	√	×	×	×	×	√	√	×	√	√
Proposition 3	×	×	×	×	√	√	√	√	√	√
Proposition 4	×	×	×	×	√	×	√	×	×	√
Proposition 5	×	√	×	×	√	×	×	×	√	√
Proposition 6	×	×	×	×	×	×	×	×	√	√
Proposition 7	√	√	√	×	√	√	√	√	×	√
Proposition 8	√	√	√	√	√	×	√	√	√	√
Proposition 9	×	√	√	×	√	√	×	×	√	√
Proposition 10	√	×	√	×	√	√	√	√	×	√
Proposition 11	√	√	√	√	√	√	√	√	√	√
Proposition 12	√	√	√	√	√	√	√	√	√	√
Proposition 13	×	√	√	×	×	×	√	√	√	√
Proposition 14	×	×	√	×	√	√	√	√	√	√

Proposition 1. *The proposed scheme preserves user anonymity*

Proof. Suppose that the attacker has intercepted U_i's login request message $\langle DID_i, A_i, T_1 \rangle$. The attacker may then try to analyze the login request message by retrieving any static parameters from this message. However, it is not feasible to derive ID_i from the login request message because the login request message includes DID_i instead of ID_i. Thus the use of DID_i ensures that the attacker cannot acquire any information related to the user identity. □

Proposition 2. *The proposed scheme achieves mutual authentication*

Proof. In our proposed scheme, the GWN can authenticate the user by checking whether the login request message is correct, and the S_n can authenticate the GWN by checking whether the message $\langle M_i, DID_i, B_i, T_2 \rangle$ is correct. To authenticate the S_n, the GWN verifies whether the message $\langle C_i, T_3 \rangle$ received by the S_n is valid or not. Also, the U_i can authenticate the GWN by checking whether the message $\langle D_i, T_4 \rangle$ is correct. If all these verification processes are successfully finished, mutual authentication has been executed properly. □

Proposition 3. *The proposed scheme provides the session key agreement*

Proof. In our proposed scheme, the user and the sensor node can share the session key after the verification procedure. As a result of the randomness and independence of the generation of R_2 in all

sessions, the shared session key $SK = h(DID_i||h(x_s||SID_n)||R_2||T_2)$ differs for each session. Therefore, it is difficult for the attacker to compute the session key from the intercepted messages. □

Proposition 4. *The proposed scheme withstands smart card loss attacks*

Proof. Suppose smart card of U_i is stolen by the attacker, who extracts secret values $\{N_i, M_i, h(\cdot), b\}$ through the studies [38]. Even if the attacker obtains $\{N_i, M_i, h(\cdot), b\}$, the attacker cannot know the user's ID_i, because our proposed scheme does not allow the ID_i to be stored in the smart card. In addition, as the ID in the smart card is erased, our proposed scheme uses a suitable password-based checking process, instead of a vulnerable id-based checking process. □

Proposition 5. *The proposed scheme withstands off-line password guessing attacks*

Proof. Suppose that the attacker extracts all of the secret information from the smart card. To successfully carry out a password guessing attack, the attacker has to know the U_i's identity ID_i. However, in our proposed scheme, it is impossible for the attacker to obtain the ID_i. Furthermore, the guessing of two secret parameters (e.g., password, identity) is computationally infeasible in polynomial time. Thus, our proposed scheme is secure against off-line password guessing attacks. □

Proposition 6. *The proposed scheme withstands user impersonation attacks*

Proof. An attacker tries to impersonate a legal user U_i in order to deceive other parties. To start a new session, the attacker has to modify the login request message $\langle DID_i, A_i, T_1 \rangle$. In order to change these values, the attacker has to know the ID_i. However, there is no way to obtain the user's ID_i. Therefore, our proposed scheme is secure against user impersonation attacks. □

Proposition 7. *The proposed scheme quickly detects the incorrect password*

Proof. In our proposed scheme, when the user inputs the incorrect password PW_a, the smart card calculates $\overline{PW_a} = h(PW_a||b)$ and $v_a = N_i \oplus h(ID_i||\overline{PW_a})$. The smart card further computes $M_a = h(\overline{PW_a}||v_a)$ and compares it with the stored value M_i. If this condition is satisfied, the card knows the user has entered the incorrect password. However, it is obvious that M_a is not equal to M_i. Therefore, unlike Chen et al.'s scheme, the smart card can promptly detect the incorrect password at the beginning of the login phase. □

Proposition 8. *The proposed scheme withstands replay attacks*

Proof. An attacker can intercept data packets to make use of the data that is contained in some manner and can then try to login to the sensor node by using the intercepted packets that were transmitted between all parties involved. However, all messages transmitted in our proposed scheme include a current timestamp, such as T_1, T_2, T_3 or T_4. Hence, our proposed scheme can defend against replay attacks. □

Proposition 9. *The proposed scheme withstands privileged-insider attacks*

Proof. There is a possibility that a privileged insider can directly acquire the user's password from the GWN to then access the user's account in other systems by using the same password. This attack is a result of the disclosure of the user's password during the registration phase. In our proposed scheme, the U_i submits the password information to the GWN in the form of $\overline{PW_i} = h(PW_i||b)$, instead of the form PW_i. Accordingly, the privileged insider cannot acquire the user's password as an attacker. □

Proposition 10. *The proposed scheme withstands denial of service attacks*

Proof. Suppose that the attacker obtains the user's smart card, and extracts all of the information from the smart card. The attacker then tries to modify the password for denial of service attack. However, the attacker cannot change the password, because our proposed scheme uses a secure verification method at the beginning of the password change phase. To successfully pass this verification procedure, the attacker has to know the user ID_i and PW_i. Therefore, our proposed scheme is secure for denial of service attack. □

Proposition 11. *The proposed scheme withstands stolen-verifier attacks*

Proof. An attacker acquires a password-verifier from the gateway node to immediately impersonate an authenticated user. To succeed in a stolen-verifier attack, the attacker needs to know the user's password. However, as is shown in our proposed scheme, no verification table is stored in our proposed scheme. □

Proposition 12. *The proposed scheme withstands off-line identity guessing attacks*

Proof. Suppose that the attacker extracts all of the secret information from the smart card. To successfully carry out an off-line identity guessing attack, the attacker has to know user's password PW_i. However, in our proposed scheme, the attacker cannot acquire the user's password. Moreover, it is not feasible to obtain ID_i from the login request because the login request includes DID_i instead of ID_i. Therefore, the attacker does not know the user's identity in our proposed scheme. □

Proposition 13. *The proposed scheme provides a friendly and efficient password change phase*

Proof. The ideal user authentication scheme allows the user to freely change his/her password, and this should be carried out without any assistance from other parties to ensure user friendliness and efficiency. In our proposed scheme, when the user wants to change an old password, the smart card first checks the validity of the old password PW_i^{old}. If the password is valid, the user can choose the new password PW_i^{new}, and the smart card computes the new values N_i^{new} and M_i^{new}. Then smart card replaces the existing values with the new values. Thus, the password change phase for our proposed scheme is both user-friendly and effective because the user U_i does not communicate with the gateway GWN. □

Proposition 14. *The proposed scheme withstands GW-node impersonation attacks*

Proof. Suppose that the attacker obtains all transmitted message such as $\langle DID_i, A_i, T_1 \rangle$ and $\langle M_i, DID_i, B_i, T_2 \rangle$, and tries to impersonate as a legal gateway node. However, It is not feasible to decrypt the $A_i = E_k(DID_i||R_1||T_1)$ without the symmetry key k. Therefore, the attacker can not impersonate as a valid gateway node. □

5.2. Authentication Proof with BAN Logic

We prove the way in which a session key can be correctly generated between communicating parties during the authentication process using a well-known formal logic known as BAN logic [41]; BAN logic is a formal means that is widely used to analyze the security of cryptographic protocols. The basic notation for figuring out BAN logic follows below.

- $A \triangleleft S$: The A sees the sentence S.
- $A \mid\equiv S$: The sentence S is believed by A.
- $\sharp(S)$: It makes a fresh sentence S.
- $A \mid\sim S$: The A said the sentence S.

- $< S >_K$: Combine the sentence S using K.
- $A \overset{K}{\longleftrightarrow} B$: For secure communication, A and B share a secret key K.
- $A \Rightarrow S$: The sentence S is controled by A.
- $\{S\}_K$: Encrypt the sentence S using K
- $(S)_K$: Perform the hash operation to sentence X using Y.

Generally, BAN logic provides some rules as follows.

1. Message-meaning rule: $\frac{A|\equiv A \overset{K}{\leftrightarrow} B, A \lhd < S >_K}{A|\equiv B|\sim S}$: If the key K is shared between A and B, A sees the S combined by K. Then A believes that B once said S.
2. Nonce-verification rule: $\frac{A|\equiv \#(S), A|\equiv B|\sim S}{A|\equiv B|\equiv S}$: If A trusts that S is fresh and A believes B once said S, then A believes that B believes S.
3. The believe rule: $\frac{A|\equiv S, A|\equiv T}{A|\equiv (S,T)}$: If S and T are believed by A, then (S, T) are also believed by A .
4. Freshness-conjuncatenation rule: $\frac{A|\equiv \#(S)}{A|\equiv \#(S,T)}$: If freshness of S is believed by A, then A can trust the freshness of whole statement.
5. Jurisdiction rule: $\frac{A|\equiv B|\Rightarrow S, A|\equiv B|\equiv S}{A|\equiv S}$: If A establishes that B has jurisdiction over S, and A trusts that B trusts a statement S, then A also trusts S.

Our analysis based on BAN logic will fulfill the following goals:

- Goal 1. $U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$
- Goal 2. $U_i \mid\equiv GWN \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$
- Goal 3. $GWN \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$
- Goal 4. $GWN \mid\equiv U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$

Our message can be transformed into idealized form as follows:

- Message 1. $U_i \rightarrow S_n$: $\{ID_i, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n\}_{h(x_a)}$
- Message 2. $U_i \rightarrow S_n$: $\{ID_i, R_1, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_1}{\longleftrightarrow} S_n\}_{h(x_a)}$
- Message 3. $GWN \rightarrow S_n$: $(SID_n, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$
- Message 4. $GWN \rightarrow S_n$: $(SID_n, R_2, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_2}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$
- Message 5. $S_n \rightarrow GWN$: $(T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$
- Message 6. $U_i \rightarrow GWN$: $(R_1, T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)_{h(x_a)}$
- Message 7. $S_n \rightarrow U_i$: $\{R_2, T_3, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_2}{\longleftrightarrow} S_n\}_{h(x_a)}$
- Message 8. $GWN \rightarrow U_i$: $(T_4, R_2, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{h(x_a)}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)_{h(x_a)}$

We define some assumptions as follows, and these assumptions will be used in further proof.

- A1: $S_n \mid\equiv \sharp(T_1)$
- A2: $S_n \mid\equiv \sharp(T_2)$
- A3: $GWN \mid\equiv \sharp(T_3)$
- A4: $U_i \mid\equiv \sharp(T_3)$
- A5: $U_i \mid\equiv \sharp(T_4)$
- A6: $S_n \mid\equiv \sharp(R_1)$
- A7: $S_n \mid\equiv \sharp(R_2)$
- A8: $GWN \mid\equiv \sharp(R_1)$
- A9: $U_i \mid\equiv \sharp(R_2)$
- A10: $U_i \mid\equiv (U_i \overset{h(x_a)}{\longleftrightarrow} S_n)$
- A11: $S_n \mid\equiv (U_i \overset{h(x_a)}{\longleftrightarrow} S_n)$
- A12: $S_n \mid\equiv (GWN \overset{h(x_s||SID_n)}{\longleftrightarrow} S_n)$

- A13: $GWN \mid\equiv (GWN \overset{h(x_a)}{\longleftrightarrow} S_n)$
- A14: $GWN \mid\equiv (GWN \overset{h(x_s||SID_n)}{\longleftrightarrow} S_n)$
- A15: $S_n \mid\equiv U_i \Rightarrow (U_i \overset{ID_i}{\longleftrightarrow} S_n)$
- A16: $S_n \mid\equiv U_i \Rightarrow (U_i \overset{R_1}{\longleftrightarrow} S_n)$
- A17: $S_n \mid\equiv GWN \Rightarrow (GWN \overset{SID_n}{\longleftrightarrow} S_n)$
- A18: $S_n \mid\equiv GWN \Rightarrow (GWN \overset{R_2}{\longleftrightarrow} S_n)$
- A19: $GWN \mid\equiv S_n \Rightarrow (GWN \overset{R_1}{\longleftrightarrow} S_n)$
- A20: $U_i \mid\equiv S_n \Rightarrow (U_i \overset{R_2}{\longleftrightarrow} S_n)$
- A21: $GWN \mid\equiv U_i \Rightarrow (U_i \overset{SK}{\longleftrightarrow} GWN)$
- A22: $U_i \mid\equiv GWN \Rightarrow (U_i \overset{SK}{\longleftrightarrow} GWN)$

Using the BAN logic rules, idealized form, and pre-defined some assumptions, we deploy our proof as follows:

Based on Message 1, we could derive:

S1 $S_n \triangleleft \{ID_i, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n\}_{h(x_a)}$

According to the assumption A11 and the message meaning rule, we obtain:

S2 $S_n \mid\equiv U_i \mid\sim (ID_i, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n)$

According to the assumption A1 and the freshness conjuncatenation rule, we obtain:

S3 $S_n \mid\equiv \sharp(ID_i, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n)$

According to the S2, S3 and the nonce verification rule, we obtain:

S4 $S_n \mid\equiv U_i \mid\equiv (ID_i, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n)$

According to the S4 and the believe rule, we obtain:

S5 $S_n \mid\equiv U_i \mid\equiv (U_i \overset{ID_i}{\longleftrightarrow} S_n)$

According to the assumption A15 and the jurisdiction rule, we obtain:

S6 $S_n \mid\equiv (U_i \overset{ID_i}{\longleftrightarrow} S_n)$

According to the Message 2, we obtain:

S7 $S_n \triangleleft \{ID_i, R_1, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_1}{\longleftrightarrow} S_n\}_{h(x_a)}$

According to the S7, assumption A11 and the message meaning rule, we obtain:

S8 $S_n \mid\equiv U_i \mid\sim (ID_i, R_1, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_1}{\longleftrightarrow} S_n)$

According to the assumption A1, A6 and the freshness conjuncatenation rule, we obtain:

S9 $S_n \mid\equiv \sharp(ID_i, R_1, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_1}{\longleftrightarrow} S_n)$

According to the S8, S9 and the nonce verification rule, we obtain:

S10 $S_n \mid\equiv U_i \mid\equiv (ID_i, R_1, T_1, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_1}{\longleftrightarrow} S_n)$

According to the S5, S6, S10 and the believe rule, we obtain:

S11 $S_n \mid\equiv U_i \mid\equiv (U_i \overset{R_1}{\longleftrightarrow} S_n)$

According to the assumption A16 and the jurisdiction rule, we obtain:

S12 $S_n \mid\equiv (U_i \overset{R_1}{\longleftrightarrow} S_n)$

According to the Message 3, we obtain:

S13 $S_n \triangleleft (SID_n, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$

According to the S13, assumption A12 and the message meaning rule, we obtain:

S14 $S_n \mid\equiv GWN \mid\sim (SID_n, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n)$

According to the assumption A2 and the freshness conjuncatenation rule, we obtain:

S15 $S_n \mid\equiv \sharp(SID_n, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n)$

According to the S14, S15 and the nonce verification rule, we obtain:

S16 $S_n \mid\equiv GWN \mid\equiv (SID_n, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n)$

According to the S16 and the believe rule, we obtain:

S17 $S_n \mid\equiv GWN \mid\equiv (GWN \overset{SID_n}{\longleftrightarrow} S_n)$
According to the assumption A17 and the jurisdiction rule, we obtain:

S18 $S_n \mid\equiv (GWN \overset{SID_n}{\longleftrightarrow} S_n)$
According to the Message 4, we obtain:

S19 $S_n \triangleleft (SID_n, R_2, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_2}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$
According to the S19, assumption A12 and the message meaning rule, we obtain:

S20 $S_n \mid\equiv GWN \mid\sim (SID_n, R_2, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_2}{\longleftrightarrow} S_n)$
According to the assumption A2, A7 and the freshness conjuncatenation rule, we obtain:

S21 $S_n \mid\equiv \sharp(SID_n, R_2, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_2}{\longleftrightarrow} S_n)$
According to the S20, S21 and the nonce verification rule, we obtain:

S22 $S_n \mid\equiv GWN \mid\equiv (SID_n, R_2, T_2, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_2}{\longleftrightarrow} S_n)$
According to the S17, S18, S21 and the believe rule, we obtain:

S23 $S_n \mid\equiv GWN \mid\equiv (GWN \overset{R_2}{\longleftrightarrow} S_n)$
According to the assumption A18 and the jurisdiction rule, we obtain:

S24 $S_n \mid\equiv (GWN \overset{R_2}{\longleftrightarrow} S_n)$
According to the Message 5, we obtain:

S25 $GWN \triangleleft (T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n)_{h(x_s||SID_n)}$
According to the S25, assumption A14 and the message meaning rule, we obtain:

S26 $GWN \mid\equiv S_n \mid\sim (T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n)$
According to the assumption A3 and the freshness conjuncatenation rule, we obtain:

S27 $GWN \mid\equiv \sharp(T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n)$
According to the S26, S27 and the nonce verification rule, we obtain:

S28 $GWN \mid\equiv S_n \mid\equiv (T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n)$
According to the S17, S18, S28 and the believe rule, we obtain:

S29 $GWN \mid\equiv S_n \mid\equiv (GWN \overset{R_1}{\longleftrightarrow} S_n)$
According to the assumption A19 and the jurisdiction rule, we obtain:

S30 $GWN \mid\equiv (GWN \overset{R_1}{\longleftrightarrow} S_n)$
According to the Message 6, we obtain:

S31 $GWN \triangleleft (R_1, T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)_{h(x_a)}$
According to the S31, assumption A13 and the message meaning rule, we obtain:

S32 $GWN \mid\equiv U_i \mid\sim (R_1, T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the assumption A3, A8 and the freshness conjuncatenation rule, we obtain:

S33 $GWN \mid\equiv \sharp(R_1, T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the S32, S33 and the nonce verification rule, we obtain:

S34 $GWN \mid\equiv U_i \mid\equiv (R_1, T_3, GWN \overset{SID_n}{\longleftrightarrow} S_n, GWN \overset{R_1}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the S17, S18, S29, S34 and the believe rule, we obtain:

S35 $GWN \mid\equiv U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$ **(Goal 4.)**
According to the assumption A21 and the jurisdiction rule, we obtain:

S36 $GWN \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$ **(Goal 3.)**
According to the Message 7, we obtain:

S37 $U_i \triangleleft \{R_2, T_3, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_2}{\longleftrightarrow} S_n\}_{h(x_a)}$
According to the S37, assumption A10 and the message meaning rule, we obtain:

S38 $U_i \mid\equiv S_n \mid\sim (R_2, T_3, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_2}{\longleftrightarrow} S_n)$
According to the assumption A4, A9 and the freshness conjuncatenation rule, we obtain:

S39 $U_i \mid\equiv \sharp(R_2, T_3, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_2}{\longleftrightarrow} S_n)$
According to the S38, S39 and the nonce verification rule, we obtain:

S40 $U_i \mid\equiv S_n \mid\equiv (R_2, T_3, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{R_2}{\longleftrightarrow} S_n)$
According to the S5, S6, S39 and the believe rule, we obtain:

S41 $U_i \mid\equiv S_n \mid\equiv (U_i \overset{R_2}{\longleftrightarrow} S_n)$
According to the assumption A20, S41 and the jurisdiction rule, we obtain:

S42 $U_i \mid\equiv (U_i \overset{R_2}{\longleftrightarrow} S_n)$
According to the Message 8, we obtain:

S43 $U_i \triangleleft (T_4, R_2, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{h(x_a)}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)_{h(x_a)}$
According to the S43, assumption A10 and the message meaning rule, we obtain:

S44 $U_i \mid\equiv S_j \mid\sim (T_4, R_2, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{h(x_a)}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the assumption A5, A9 and the freshness conjuncatenation rule, we obtain:

S45 $U_i \mid\equiv \sharp(T_4, R_2, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{h(x_a)}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the S44, S45 and the nonce verification rule, we obtain:

S46 $U_i \mid\equiv GWN \mid\equiv (T_4, R_2, U_i \overset{ID_i}{\longleftrightarrow} S_n, U_i \overset{h(x_a)}{\longleftrightarrow} S_n, U_i \overset{SK}{\longleftrightarrow} GWN)$
According to the S5, S6, S41, S46 and the believe rule, we obtain:

S47 $U_i \mid\equiv GWN \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$ **(Goal 2.)**
According to the assumption A22 and the jurisdiction rule, we obtain:

S48 $U_i \mid\equiv (U_i \overset{SK}{\longleftrightarrow} GWN)$ **(Goal 1.)**

Based on (Goal 1–Goal 4), we can assure that our proposed scheme provides the mutual authentication and agreement of the session key SK, which is correctly shared between U_i and GWN.

6. Performance Analysis of the Proposed Scheme

In this section, we summarize the performance analysis of our proposed scheme in terms of the computation and communication complexities. These two factors are the most important when measuring the performance of any user authentication and key agreement protocol for WSN, and it would be more efficient for the complexities to be less than that of existing schemes. We thus present a performance evaluation to compare our proposed scheme to other related schemes [13–20,28].

6.1. Computational Performance Analysis

In this subsection, we present a comparison of the computational costs, and measure the execution time. The computational analysis of an authentication protocol is generally conducted by focusing on operations performed by each party within the protocols. Therefore, for analysis of the computational costs, we concentrated on the operations that are conducted by the parties in WSNs: namely a user, a gateway node, a sensor node, and a base station. A base station is used to gather the information detected by sensor node or gateway node. Our scheme also analyzes the messages which are delivered in each communication party within the protocols. This analysis of the message size is relevant to the communication cost, and there are more details in Section 6.2. In order to facilitate the analysis of the computational costs, we define the following notation.

- T_H: the time to execute a one-way hashing operation
- $T_{E/D}$: the time to compute a symmetric-key encryption/decryption
- T_{ECC}: the time to compute an encryption/decryption operation in ECC-160 algorithm

In addition, in order to achieve accurate measurement, we performed an experiment. This experiment was performed using the Crypto++ Library [42] on a system using the 64-bits Windows 7 operating system, 3.2 GHz processor, 4 GB memory, Visual C++ 2013 Software, the SHA-1 hash function, the AES symmetric encryption/decryption function, and the ECC-160 function. According to our experiment, T_H is nearly 0.0002 s on average, $T_{E/D}$ is nearly 0.0087 s on average and T_{ECC} is nearly 0.6 s on average.

Table 3 compiles a comparative analysis of the computational cost among the related schemes [13–20,28]. For example to calculate computational costs, the computation costs of sensor node are $3T_H$ from our proposed scheme in Table 3. Sensor node is the sum of three values from hash operation, $SK = h(DID_i||h(x_s||SID_n)||R_2||T_2)$, $B_i^* = h(DID_i||SK||h(x_s||SID_n)||SID_n||T_2)$, and $C_i = h(h(x_s||SID_n)||SK||DID_i||SID_n||T_3)$, in login and verification phase. However, the value of $h(x_s||SID_n)$ is not counted, since it is already contained in sensor node. Using this computation method, we analyze by comparing the computational load during the login and verification phases. Table 3 shows that Yeh et al.'s scheme [18] imposes the highest computational load, because their scheme uses an ECC operation. In contrast with Chen et al.'s scheme [28], the total computational costs for the proposed scheme uses only three more hash operations. However, there is almost no difference between them in terms of computational complexities, because the hash function is an extremely lightweight operation. In addition, even though our proposed scheme is more computationally costly than some of the other schemes, this should be easily tolerated because our proposed scheme assures higher security, and affords resistance to most well known attacks, while providing functionality.

Table 3. Comparison of the computational cost between our proposed scheme and other related schemes.

Schemes	User	Gateway Node	Sensor Node	Base Station	Total
Proposed scheme	$5T_H + 2T_{E/D}$	$5T_H + 2T_{E/D}$	$3T_H$	-	$13T_H + 4T_{E/D}$
Chen et al. [28]	$2T_H + 2T_{E/D}$	$5T_H + 2T_{E/D}$	$3T_H$	-	$10T_H + 4T_{E/D}$
Xue et al. [20]	$7T_H$	$13T_H$	$6T_H$	-	$26T_H$
Das et al. [19]	$5T_H + 1T_{E/D}$	$2T_H + 2T_{E/D}$	-	$3T_H + 3T_{E/D}$	$10T_H + 6T_{E/D}$
Yeh et al. [18]	$1T_H + 2T_{ECC}$	$4T_H + 2T_{ECC}$	$3T_H + 2T_{ECC}$	-	$8T_H + 6T_{ECC}$
Fan et al. [17]	$7T_H$	$8T_H$	$2T_H$	$2T_H$	$19T_H$
C-S- [16]	$4T_H$	$5T_H$	$1T_H$	-	$10T_H$
Vaidya et al. [15]	$6T_H$	$5T_H$	$2T_H$	-	$13T_H$
K-A- [14]	$4T_H$	$6T_H$	$2T_H$	-	$12T_H$
Das et al. [13]	$4T_H$	$1T_H$	$4T_H$	-	$9T_H$

Table 4 presents the time consumption of the proposed scheme and the other related schemes [13–20,28]. Most of authentication researches [24,29,43–45] use the following ways to compute execution time of protocol: (1) calculate protocol's computational costs, (2) measure each operation's execution time by simulation, and (3) apply the execution time derived by (2) into (1). The values on Table 4 are also based on total computational costs derived by Table 3. That is, the values of simulations ($T_H \approx 0.0002$, $T_{E/D} \approx 0.0087$, $T_{ECC} \approx 0.6$) are substituted into the total computational costs on Table 3. Total computational costs of proposed scheme are $13T_H + 4T_{E/D}$, which is ($13 \times 0.002 + 4 \times 0.0087 \approx 0.0374$ s). Other scheme's execution times are compared in the same way: Das et al. [13] ($9T_H \approx 9 \times 0.0002$), K-A- [14] ($12T_H \approx 12 \times 0.0002$), Vaidya et al. [15] ($13T_H \approx 13 \times 0.0002$), C-S- [16] ($10T_H \approx 10 \times 0.0002$), Fan et al. [17] ($19T_H \approx 19 \times 0.0002$), Yeh et al. [18] ($8T_H + 6T_{ECC} \approx 8 \times 0.0002 + 6 \times 0.6$), Das et al. [19] ($10T_H + 6T_{E/D} \approx 10 \times 0.0002 + 6 \times 0.0087$), Xue et al. [20] ($26T_H \approx 26 \times 0.0002$), Chen et al. [28] ($10T_H + 4T_{E/D} \approx 10 \times 0.0002 + 4 \times 0.0087$) are 0.0018 s, 0.0024 s, 0.0026 s, 0.002 s, 0.0038 s, 3.6016 s, 0.0542 s, 0.0052 s, 0.0368 s, respectively. Table 4 shows that the execution time of our proposed scheme is only 0.0374 s, so it can be regarded as of negligible significance. Whereas, Yeh et al.'s scheme [18] using ECC operation requires 3.6016 s, and therefore Yeh et al.'s scheme turns out to be ineffective. There is no need for concern about the execution time difference between our scheme and the other systems. The Table 4 shows our scheme takes slightly more time, but it is hard for the users to perceive this time difference. From Tables 3 and 4, we conclude that our proposed scheme considers the efficiency.

Table 4. Comparison of the execution times.

Das's [13]	K-A-'s [14]	Vaidya's [15]	C-S-'s [16]	Fan's [17]	Yeh's [18]	Das's [19]	Xue's [20]	Chen's [28]	Proposed Scheme
≈0.0018 s	≈0.0024 s	≈0.0026 s	≈0.002 s	≈0.0038 s	≈3.6016 s	≈0.0542 s	≈0.0052 s	≈0.0368 s	≈0.0374 s

6.2. Communication Performance Analysis

In this subsection, we analyze the messages that are delivered to each party within the protocols. This analysis of the message size is relevant to the communication cost. We compare the number of messages and the total number of bytes for all messages to be transmitted during the login and verification phases. Table 5 shows the communication cost between our proposed scheme and the other schemes [13–20,28]. We have analyzed all the schemes mentioned in Table 5, and the details of algorithms of related works [13–20] are described in Appendixes A–H. Based on [23], we set that both the block size of the symmetric encryption and one-way hash function $h(\cdot)$ are 20 bytes long, the identity ID_i and password PW_i are 8 bytes, the random number b, R_1, and R_2 are 16 bytes, the timestamp T_1–T_4 are 19 bytes, and ECC function is 15 bytes long. Table 5 shows that in Chen et al.'s scheme [28], the login request message $\langle ID_i, A_i, T_1 \rangle$ requires (8 + 20 + 19) = 47 bytes, and the authentication message $\langle ID_i, B_i, T_2 \rangle$ requires (8 + 20 + 19) = 47 bytes. The last two authentication messages $\langle C_i, T_3 \rangle$ and $\langle D_i, T_4 \rangle$ require (20 + 19) = 39 bytes and (20 + 19) = 39 bytes, respectively. Thus, their scheme requires a total of 172 bytes.

Table 5. Comparison of the communication cost between our proposed scheme and other related schemes.

Schemes	Total Number of Messages Required	Total Number of Bytes Required
Proposed scheme	4 Messages	216 Bytes
Chen et al. [28]	4 Messages	172 Bytes
Xue et al. [20]	6 Messages	284 Bytes
Das et al. [19]	4 Messages	253 Bytes
Yeh et al. [18]	3 Messages	118 Bytes
Fan et al. [17]	3 Messages	126 Bytes
Chen and Shih [16]	4 Messages	170 Bytes
Vaidya et al. [15]	5 Messages	157 Bytes
Khan and Alghathbar [14]	4 Messages	157 Bytes
Das et al. [13]	3 Messages	118 Bytes

In our proposed scheme, the login request message $\langle DID_i, A_i, T_1 \rangle$ requires (20 + 20 + 19) = 59 bytes, and the authentication message $\langle M_i, DID_i, B_i, T_2 \rangle$ requires (20 + 20 + 20 + 19) = 79 bytes. The second authentication message $\langle C_i, T_3 \rangle$ requires (20 + 19) = 39 bytes, and the third authentication message $\langle D_i, T_4 \rangle$ requires (20 + 19) = 39 bytes. Adding all these together, the communication overhead becomes (59 + 79 + 39 + 39) = 216 bytes. Table 5 shows that our proposed scheme requires a little more communication cost than Chen et al.'s scheme [28]. However, our scheme corrects the flaws of Chen et al.'s scheme, such as smart card loss attack, and denial of service attack. Also, even though our scheme requires a little more communication cost than some of the other schemes, we consider this acceptable because our proposed scheme assures security and provides additional functionalities, as Table 2 shows.

7. Conclusions

In this study, we analyze the security weaknesses of Chen et al.'s scheme, and show that their scheme is susceptible to smart card loss attack and denial of service attack. In addition, we also show that Chen et al.'s scheme cannot preserve user anonymity, and their scheme cannot quickly detect an incorrect password during the login phase. So, we propose a security enhanced user authentication and key agreement scheme using a symmetric cryptosystem for WSNs. The proposed scheme not

only preserves the merits of Chen et al.'s scheme, but also fixes its security flaws. Our security and performance comparison shows that our protocol achieves both stronger security and higher efficiency. Therefore, we estimate that our proposed scheme is more suitable for applications in WSNs.

Acknowledgments: This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT, and Future Planning (2014R1A1A2002775).

Author Contributions: J.J., J.K. and Y.C. conceived and designed the experiments; J.J. performed the experiments; J.J. and Y.C. analyzed the data; J.J. and D.W. wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Das et al.'s Authentication Scheme [13]

Das et al.'s authentication scheme is shown in Figures A1 and A2.

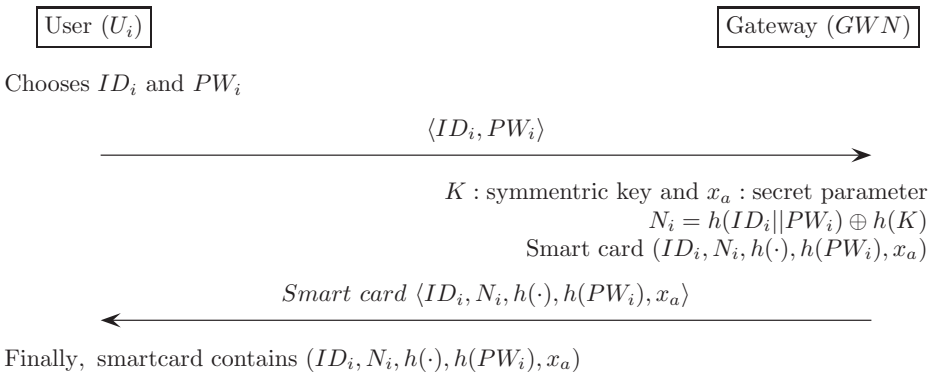

Figure A1. Registration phase for the Das et al.'s scheme [13].

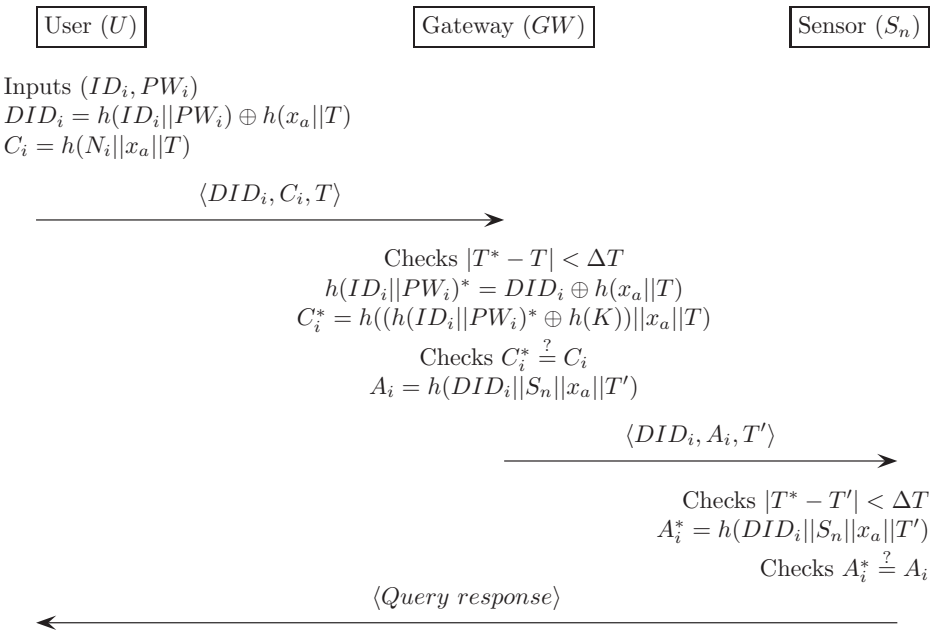

Figure A2. Login and verification phase for the Das et al.'s scheme [13].

Appendix B. Khan and Alghathbar's Authentication Scheme [14]

Khan and Alghathbar's authentication scheme is shown in Figures B1 and B2.

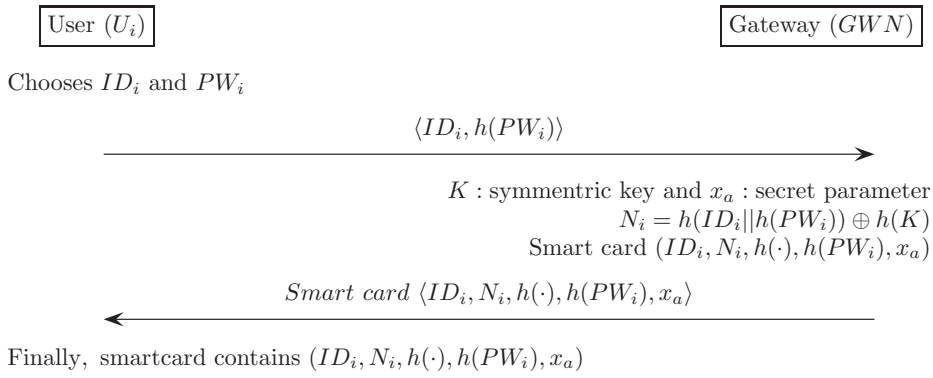

Figure B1. Registration phase for the Khan and Alghathbar's scheme [14].

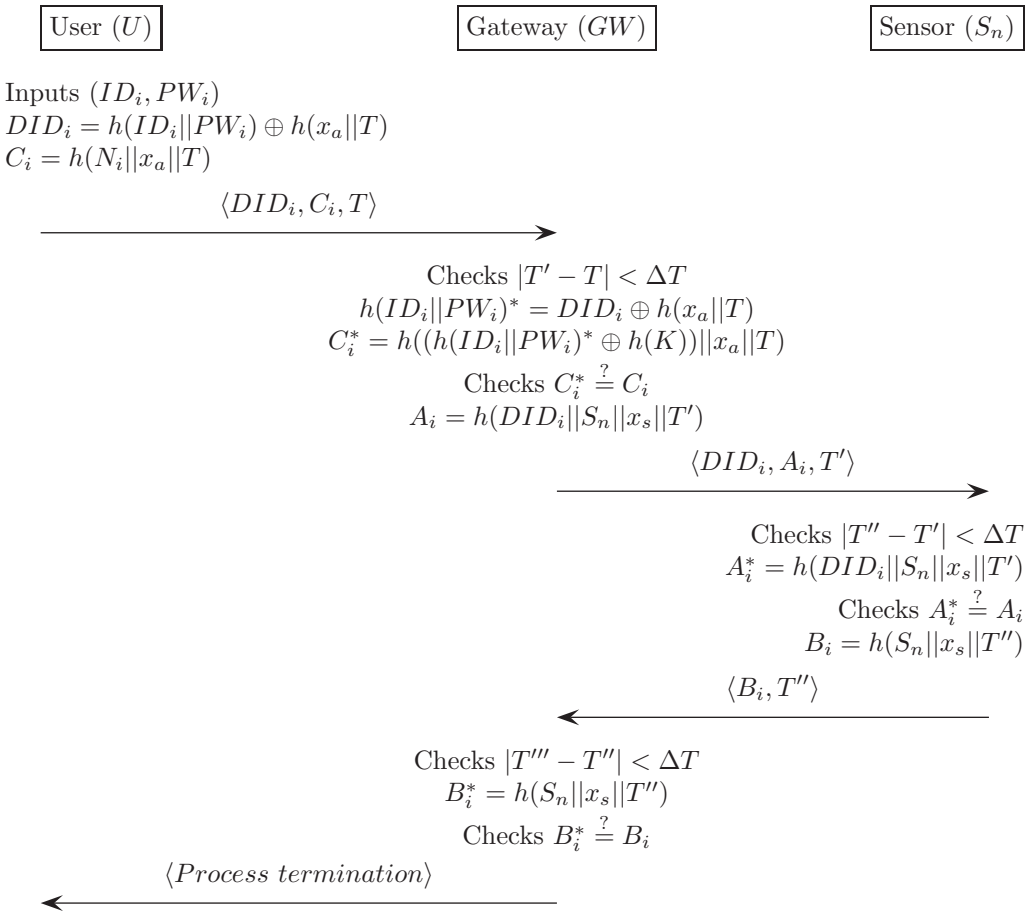

Figure B2. Login and verification phase for the Khan and Alghathbar's scheme [14].

Appendix C. Vaidya et al.'s Authentication Scheme [15]

Vaidya et al.'s authentication scheme is shown in Figures C1 and C2.

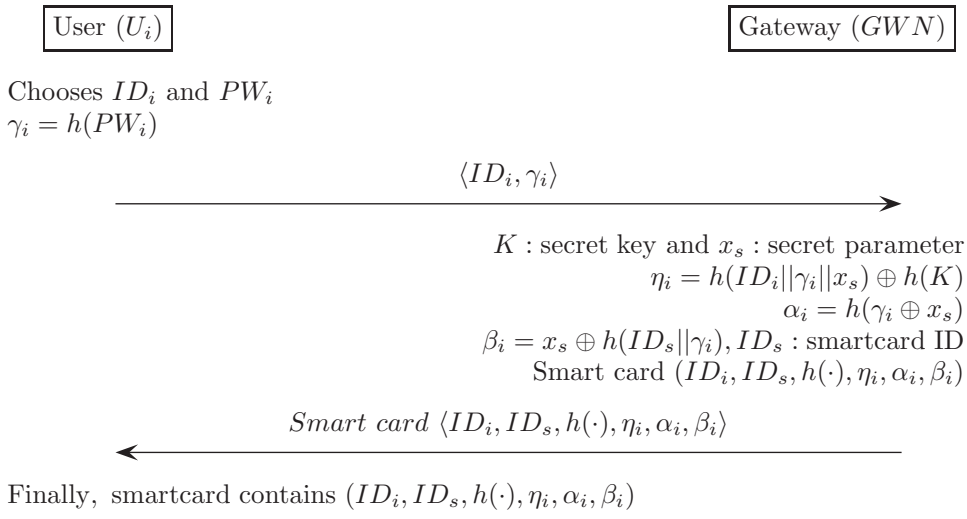

Figure C1. Registration phase for the Vaidya et al.'s scheme [15].

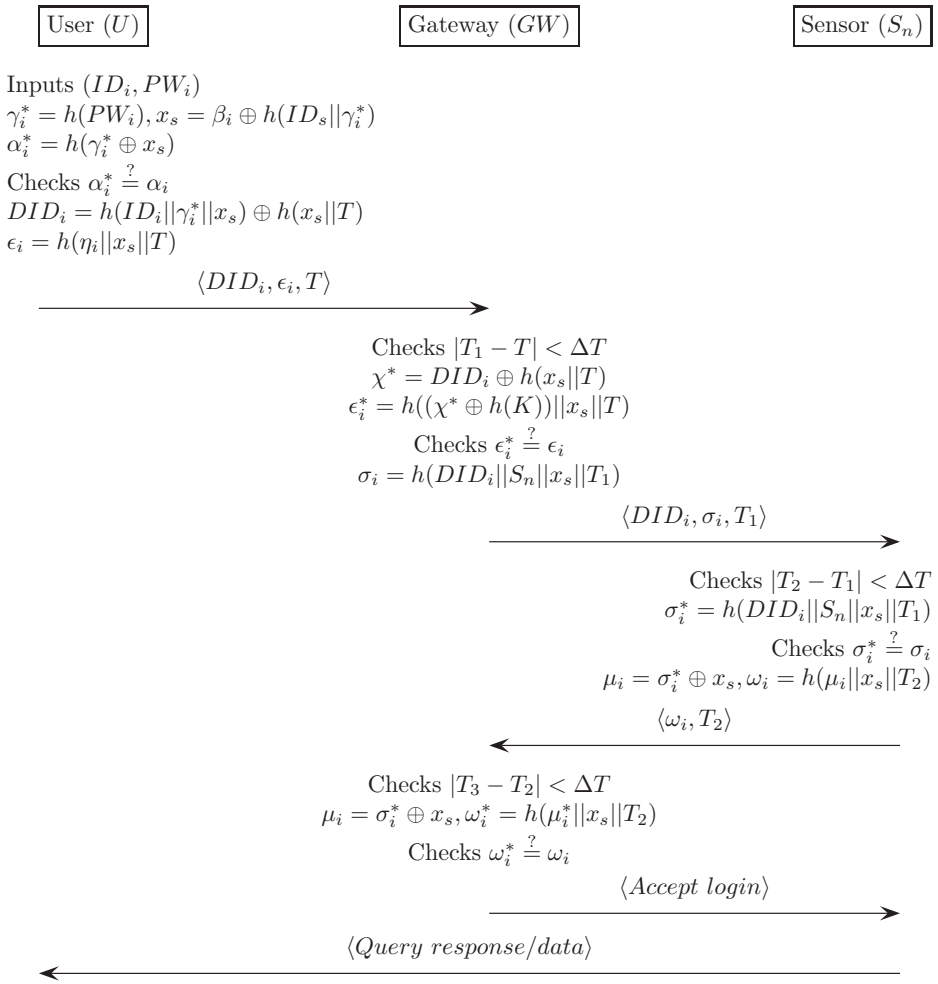

Figure C2. Login and verification phase for the Vaidya et al.'s scheme [15].

Appendix D. Chen and Shih's Authentication Scheme [16]

Chen and Shih's authentication scheme is shown in Figures D1 and D2.

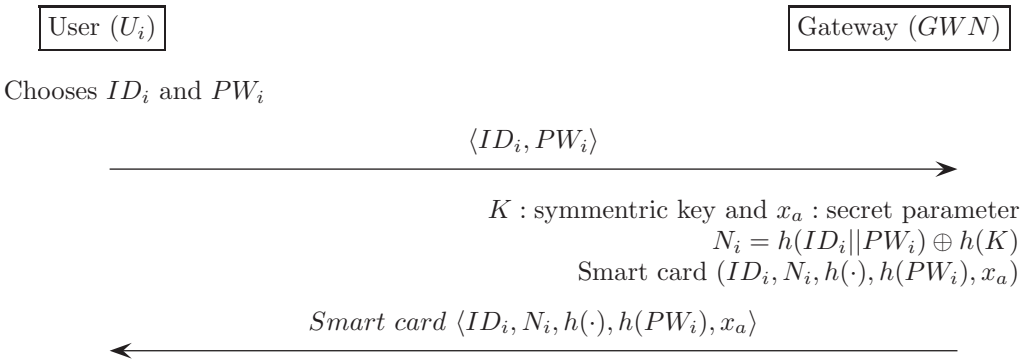

Figure D1. Registration phase for the Chen and Shih's scheme [16].

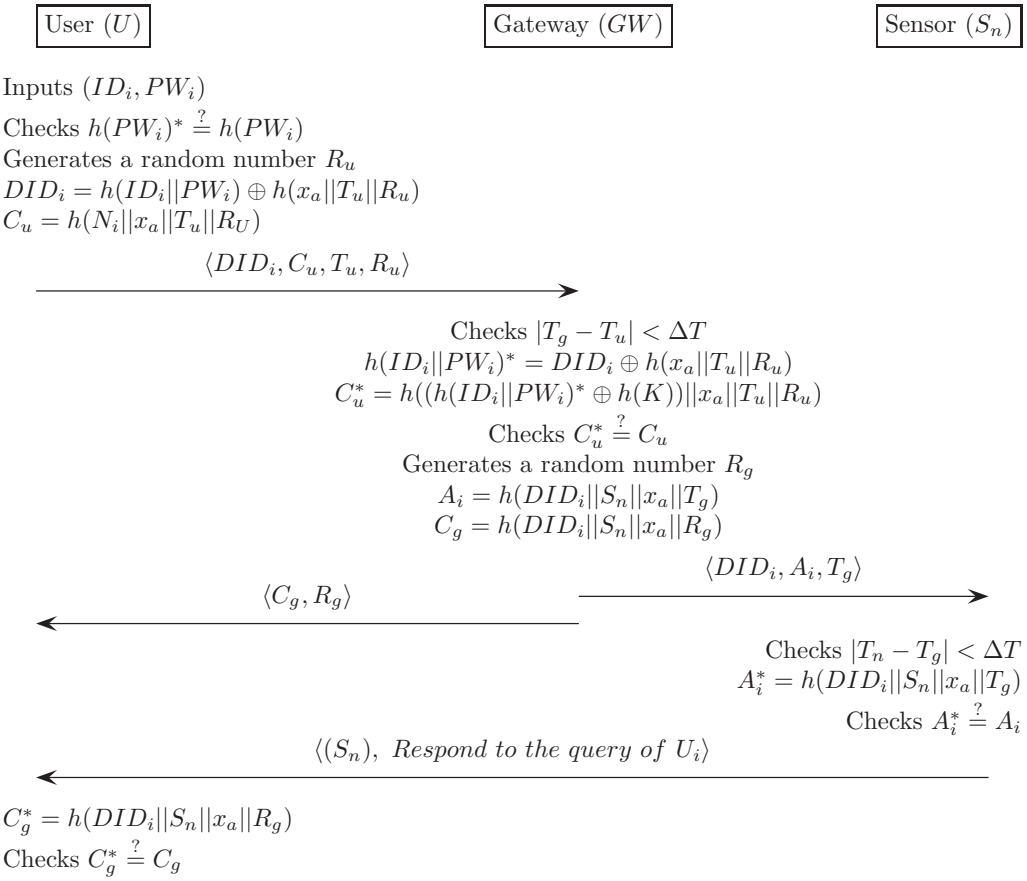

Figure D2. Login and verification phase for the Chen and Shih's scheme [16].

Appendix E. Fan et al.'s Authentication Scheme [17]

Fan et al.'s authentication scheme is shown in Figures E1 and E2.

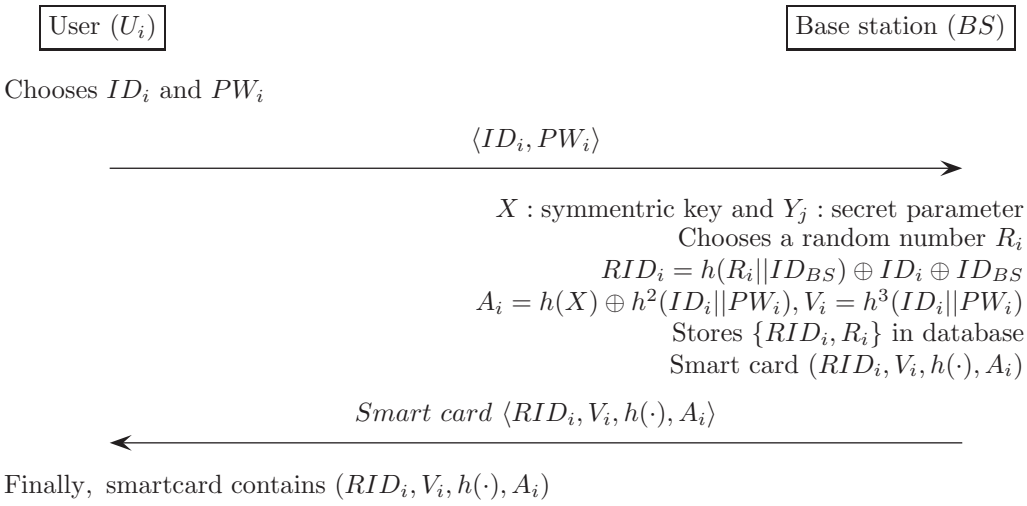

Figure E1. Registration phase for the Fan et al.'s scheme [17].

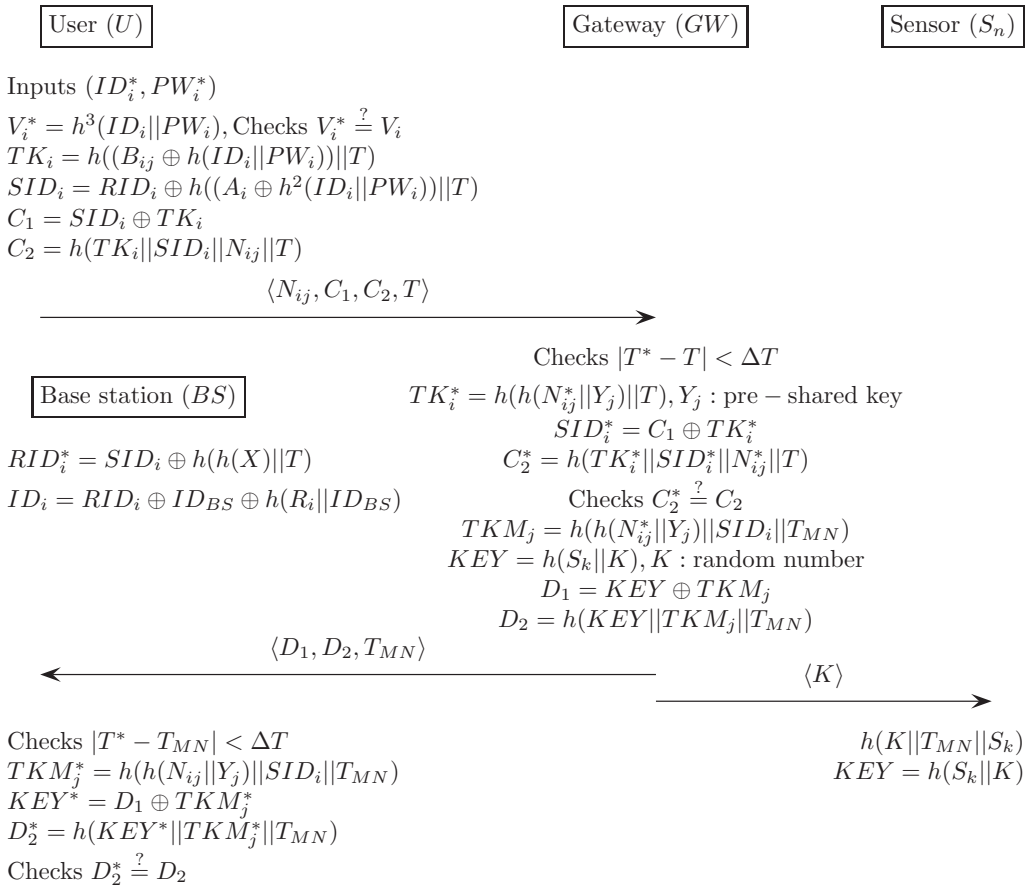

Figure E2. Login and verification phase for the Fan et al.'s scheme [17].

Appendix F. Yeh et al.'s Authentication Scheme [18]

Yeh et al.'s authentication scheme is shown in Figures F1 and F2.

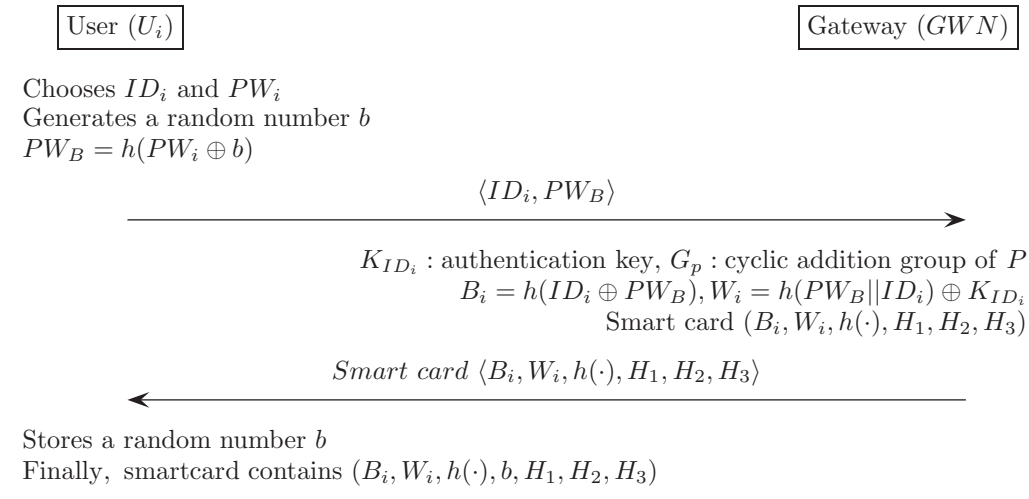

Figure F1. Registration phase for the Yeh et al.'s scheme [18].

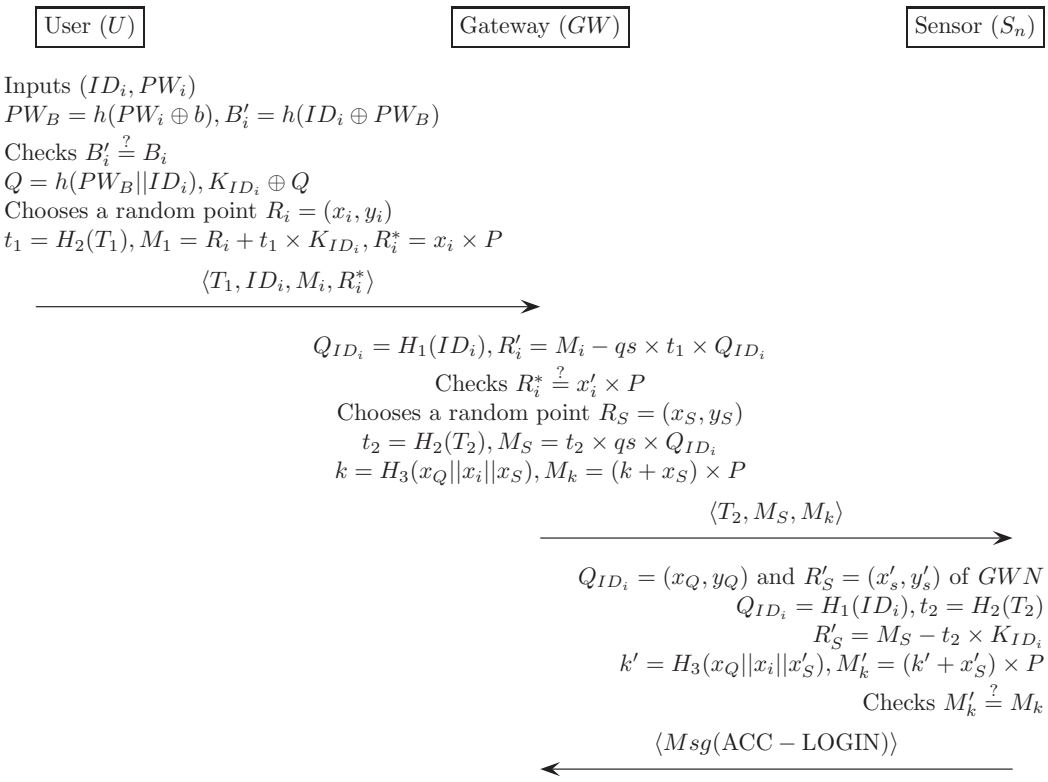

Figure F2. Login and verification phase for the Yeh et al.'s scheme [18].

Appendix G. Das et al.'s Authentication Scheme [19]

Das et al.'s authentication scheme is shown in Figures G1 and G2.

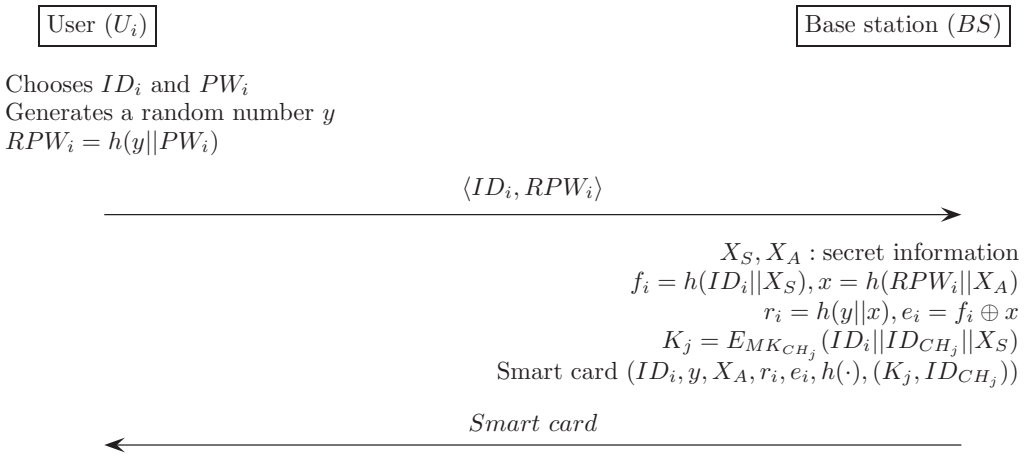

Figure G1. Registration phase for the Das et al.'s scheme [19].

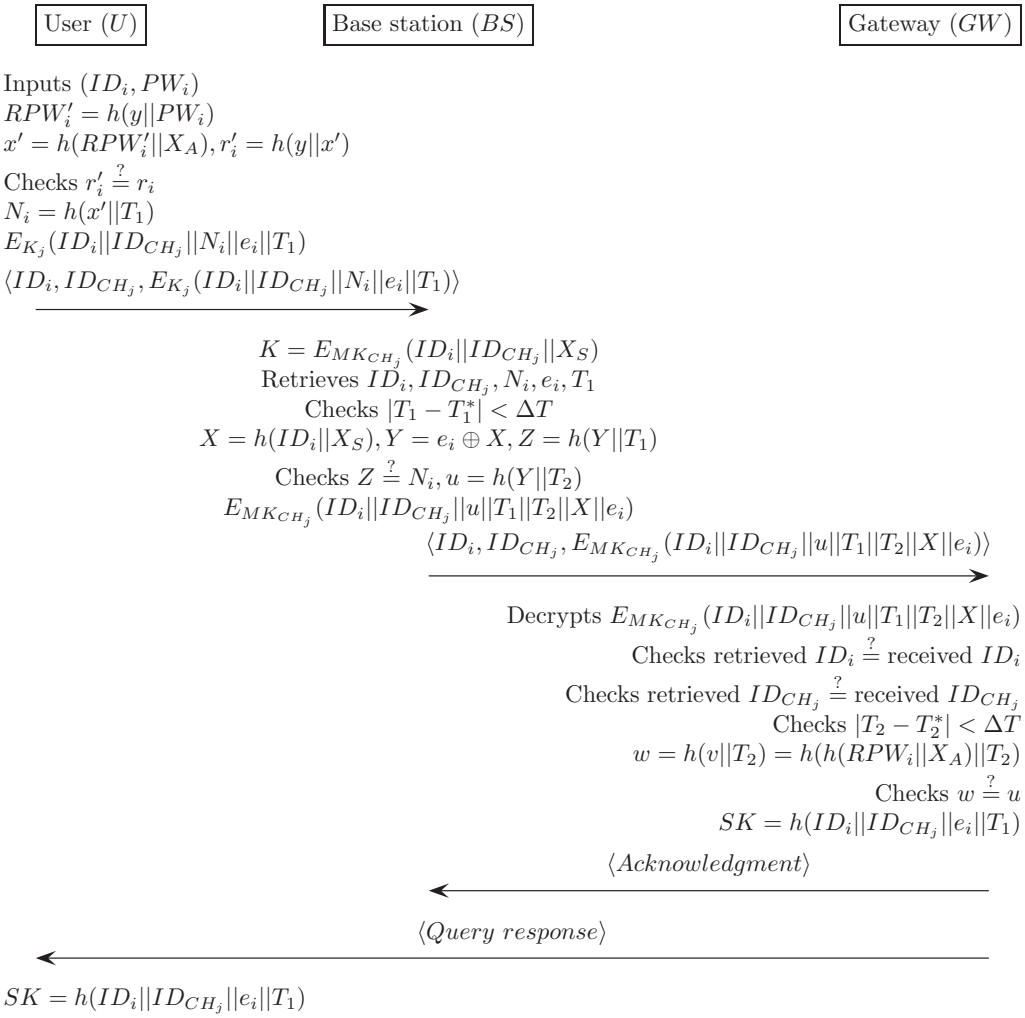

Figure G2. Login and verification phase for the Das et al.'s scheme [19].

Appendix H. Xue et al.'s Authentication Scheme [20]

Xue et al.'s authentication scheme is shown in Figures H1 and H2.

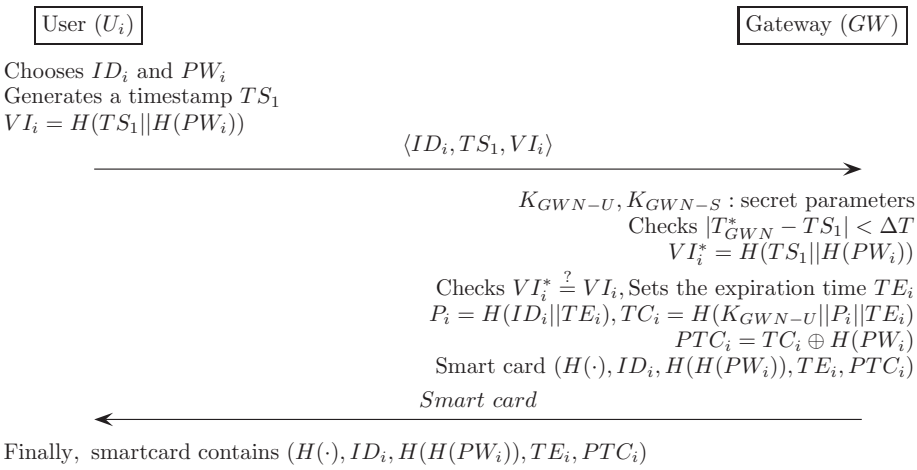

Figure H1. Registration phase for the Xue et al.'s scheme [20].

Figure H2. Login and verification phase for the Xue et al.'s scheme [20].

References

1. Yick, J.; Mukherjee, B.; Ghosal, D. Wireless sensor network survey. *Comput. Netw.* **2008**, *52*, 2292–2330.
2. Chong, C.Y.; Kumar, S.P. Sensor networks: evolution, opportunities, and challenges. *Proc. IEEE*. **2003**, *91*, 1247–1256.
3. Claycomb, W.R.; Shin, D. A novel node level security policy framework for wireless sensor networks. *J. Netw. Comput. Appl.* **2011**, *34*, 418–428.
4. Rivest, R.L.; Shamir, A.; Adleman, L. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* **1978**, *21*, 120–126.
5. Watro, R.; Kong, D.; Cuti, S.F.; Gardiner, C.; Lynn, C.; Kruus, P. TinyPK: Securing sensor networks with public key technology. In Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks, Washington, DC, USA, 25 October 2004; pp. 59–64.
6. Koblitz, N. Elliptic curve cryptosystems. *Math. Comput.* **1987**, *48*, 203–209.
7. ElGamal, T. A public key cryptosystem and a signature scheme based on discrete logarithms. In *Workshop on the Theory and Application of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1984; pp. 10–18.
8. Hwang, M.S.; Li, L.H. A new remote user authentication scheme using smart cards. *IEEE Trans. Consum. Electron.* **2000**, *46*, 28–30.
9. Lamport, L. Password authentication with insecure communication. *Commun. ACM* **1981**, *24*, 770–772.
10. Wong, K.H.; Zheng, Y.; Cao, J.; Wang, S. A dynamic user authentication scheme for wireless sensor networks. In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, Taichung, Taiwan, 5–7 June 2006; Volume 1, pp. 1–9.
11. Tseng, H.R.; Jan, R.H.; Yang, W. An Improved dynamic user authentication scheme for wireless sensor networks. In Proceedings of the Global Telecommunications Conference, Washington, DC, USA, 26–30 November 2007.
12. Vaidya, B.; SáSilva, J.; Rodrigues, J.J.P.C. Robust dynamic user authentication scheme for wireless sensor networks. In Proceedings of the 5th ACM Symposium on QoS and Security for Wireless and Mobile Networks, New York, NY, USA, 28 October 2009; pp. 88–91.
13. Das, M.L. Two-factor user authentication scheme in wireless sensor networks. *IEEE Trans. Wirel. Commun*. **2009**, *8*, 1086–1090.
14. Khan, M.K.; Alghathbar, K. Cryptanalysis and security improvements of two-factor user authentication in wireless sensor networks. *Sensors* **2010**, *10*, 2450–2459.
15. Vaidya, B.; Makrakis, D.; Mouftah, H.T. Improved two-factor user authentication in wireless sensor networks. In Proceedings of the IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Niagara Falls, ON, USA, 11–13 October 2010; pp. 600–606.
16. Chen, T.H.; Shih, W.K. A Robust Mutual Authentication Protocol for Wireless Sensor Networks. *ETRI J.* **2010**, *32*, 704–712.
17. Fan, R.; Ping, L.D.; Fu, J.Q.; Pan, X.Z. A secure and efficient user authentication protocol for two-tiered wireless sensor networks. In Proceedings of the 2010 Second Pacific-Asia Conference on Circuits, Communications and System (PACCS), Beijing, China, 1–2 August 2010; Volume 1, pp. 425–428.
18. Yeh, H.L.; Chen, T.H.; Liu, P.C.; Kim, T.H.; Wei, H.W. A secured authentication protocol for wireless sensor networks using elliptic curves cryptography. *Sensors* **2011**, *11*, 4767–4779.
19. Das, A.K.; Sharma, P.; Chatterjee, S.; Sing, J.K. A dynamic password-based user authentication scheme for hierarchical wireless sensor networks. *J. Netw. Comput. Appl.* **2012**, *35*, 1646–1656.
20. Xue, K.; Ma, C.; Hong, P.; Ding, R. A temporal-credential-based mutual authentication and key agreement scheme for wireless sensor networks. *J. Netw. Comput. Appl.* **2013**, *36*, 316–323.
21. Yuan, J.J. An enhanced two-factor user authentication in wireless sensor networks. *Telecommun. Syst.* **2014**, *55*, 105–113.
22. Turkanović, M.; Brumen, B.; Hölbl, M. A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion. *Ad Hoc Netw.* **2014**, *20*, 96–112.
23. Farash, M.S.; Turkanović, M.; Kumari, S.; Hölbl, M. An efficient user authentication and key agreement scheme for heterogeneous wireless sensor network tailored for the Interne tof Things environment. *Ad Hoc Netw.* **2014**, 36, 152–176.

24. Amin, R.; Islam, S.H.; Biswas, G.P.; Khan, M.K.; Leng, L.; Kumar, N. Design of an anonymity-preserving three-factor authenticated key exchange protocol for wireless sensor networks. *Comput. Netw.* **2016**, *101*, 42–62.
25. Kothmayr, T.; Schmitt, C.; Hu, W.; Brünig, M.; Carle, G. DTLS based security and two-way authentication for the Internet of Things. *Ad. Hoc. Netw.* **2013**, *11*, 2710–2723.
26. Schmitt, C.; Noack, M.; Stiller, B. Chapter 13: TinyTO: Two-way Authentication for Constrained Devices in the Internet-of-Things. In *Internet-of-Things (Principles and Paradigms)*; Morgen Kaufmann: Cambridge, MA, USA, 2016; pp. 239–258.
27. Porambage, P.; Schmitt, C.; Kumar, P.; Gurtov, A.; Ylianttila, M. Pauthkey: A pervasive authentication protocol and key establishment scheme for wireless sensor networks in distributed IoT applications. *Int. J. Distrib. Sens. Netw.* **2014**, *2014*, doi:10.1155/2014/357430.
28. Chen, L.; Wei, F.; Ma, C. A secure user authentication scheme against smart-card loss attack for wireless sensor networks using symmetric key techniques. *Int. J. Distrib. Sens. Netw.* **2015**, *2015*, doi:10.1155/2015/704502.
29. Kang, D.; Jung, J.; Mun, J.; Lee, D.; Choi, Y.; Won, D. Efficient and robust user authentication scheme that achieve user anonymity with a Markov chain. *Secur. Commun. Netw.* **2016**, *9*, doi:10.1002/sec.1432.
30. Syverson, P. A taxonomy of replay attacks [cryptographic protocols]. In Proceedings of the Computer Security Foundations Workshop VII, CSFW 7, Franconia, VA, USA, 14–16 June 2014; pp. 187–191.
31. Chien-Ming, C.; Wei-Chi, K. Stolen-verifier attack on two new strong-password authentication protocols. *IEICE Trans. Commun.* **2002**, *85*, 2519–2521.
32. Schultz, E.E. A framework for understanding and predicting insider attacks. *Comput. Secur.* **2002**, *21*, 526–531.
33. Wei-Chi, K.U.; Chang, S.T. Impersonation attack on a dynamic ID-based remote user authentication scheme using smart cards. *IEICE Trans. Commun.* **2005**, *88*, 2165–2167.
34. Gong, L. Optimal authentification protocols resistant to password guessing attacks. In Proceedings of the IEEE 8th Computer Security Foundations Workshop, County Kerry, UK, 13–15 June 1995; pp. 24–29.
35. Kim, J.; Lee, D.; Jeon, W.; Lee, Y.; Won, D. Security Analysis and Improvements of Two-Factor Mutual Authentication with Key Agreement in Wireless Sensor Networks. *Sensors* **2014**, *14*, 6443–6462.
36. Choi, Y.; Lee, D.; Kim, J.; Jung, J.; Nam, J.; Won, D. Security Enhanced User Authentication Protocol for Wireless Sensor Networks Using Elliptic Curves Cryptography. *Sensors* **2014**, *14*, 10081–10106.
37. Choi, Y.; Nam, J.; Lee, D.; Kim, J.; Jung, J.; Won, D. Security Enhanced Anonymous Multi-Server Authenticated Key Agreement Scheme Using Smart Cards and Biometrics. *Sci. World J.* **2014**, *2014*, doi:10.1155/2014/281305.
38. Kocher, P.; Jaffe, J.; Jun, B. Differential power analysis. In Proceedings of the Advances in Cryptology-CRYPTO'99, LNCS, Santa Barbara, CA, USA, 16 December 1999; Volume 1666, pp. 388–397.
39. Amin, R.; Biswas, G.P. A secure light weight scheme for user authentication and key agreement in multi-gateway based wireless sensor networks. *Ad Hoc Netw.* **2016**, *36*, 58–80.
40. Li, X.; Niu, J.; Khan, M.K.; Liao, J. An enhanced smart card based remote user password authentication scheme. *J. Netw. Comput. Appl.* **2013**, *36*, 1365–1371.
41. Burrows, M.; Abadi, M.; Needham, R.M. A logic of authentication. *Proc. R. Soc. Lond. A. Math. Phys. Sci.* **1989**, *426*, 233–271.
42. Dai, W. Crypto++ Library, 5.6.1. Available online: http://www.cryptopp.com (accessed on 5 April 2011).
43. Li, C.T.; Hwang, M.S.; Chu, Y.P. A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks. *Comput. Commun.* **2008**, *31*, 2803–2814.
44. Li, C.T.; Weng, C.Y.; Lee, C.C. An advanced temporal credential-based security scheme with mutual authentication and key agreement for wireless sensor networks. *Sensors* **2013**, *13*, 9589–9603.
45. Chang, C.C.; Le, H.D. A provably secure, efficient, and flexible authentication scheme for ad hoc wireless sensor networks. *IEEE Trans. Wirel. Commun.* **2015**, *15*, 357–366.